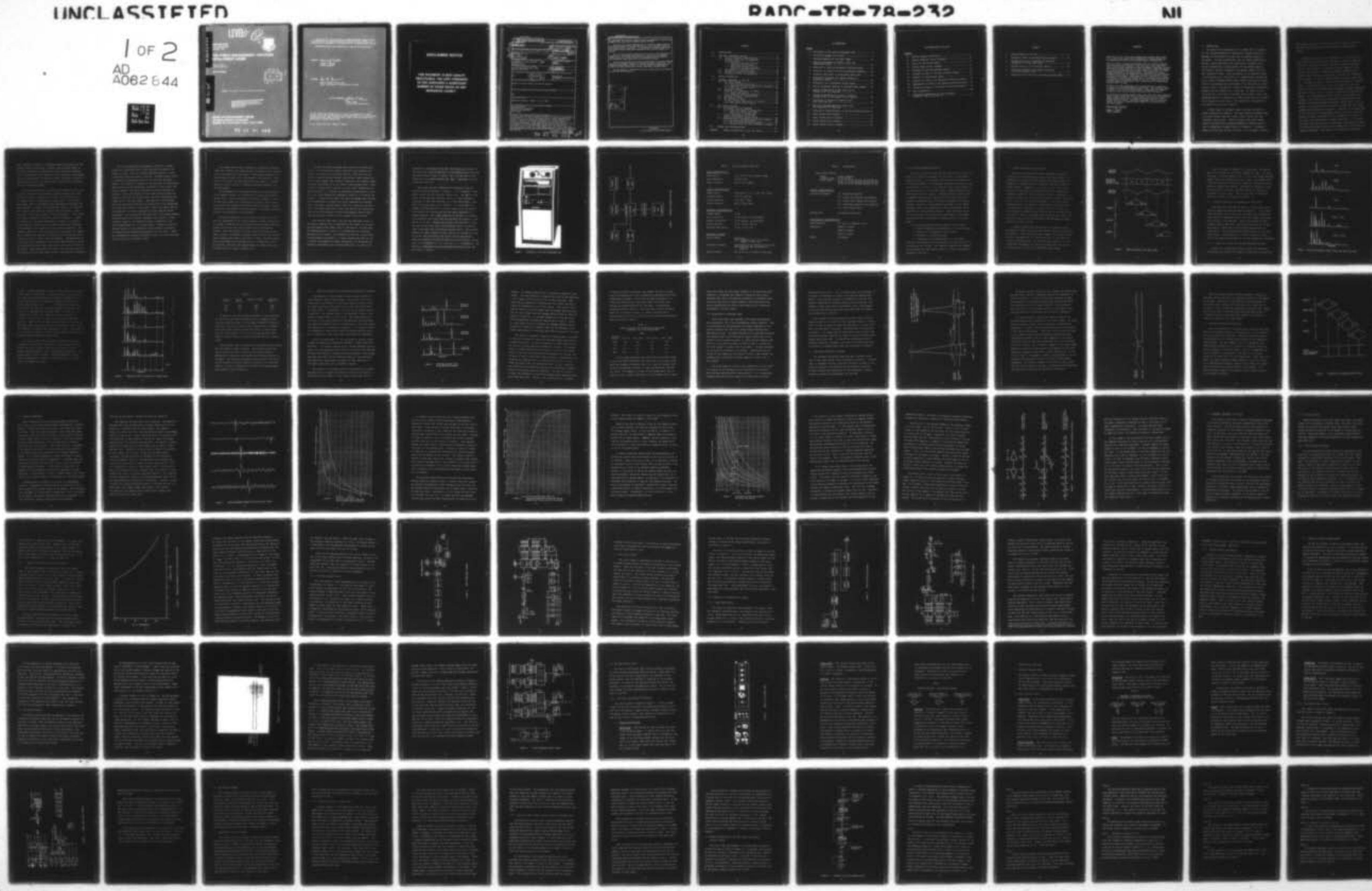
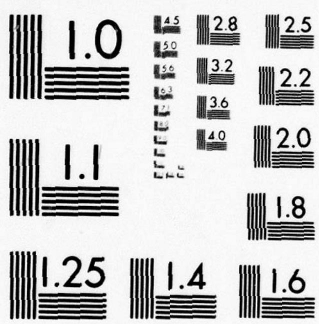
AD-A062 844 QUEENS COLL FLUSHING NY DEPT OF COMPUTER SCIENCES F/G 17/2
THE SPEECH ENHANCEMENT ADVANCED DEVELOPMENT MODEL.(U)
NOV 78 M R WEISS, E ASCHKENASY F30602-76-C-0033
UNCLASSIFIED RADC-TR-78-232 NL

1 OF 2
AD
A062 844





MICROCOPY RESOLUTION TEST CHART

LEVEL II

12

RADC-TR-78-232
Final Technical Report
November 1978

THE SPEECH ENHANCEMENT ADVANCED DEVELOPMENT MODEL

Mark R. Weiss
Ernest Aschkensasy

Queens College

Approved for public release; distribution unlimited

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

This report has been reviewed by the RADC Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

RADC-TR-78-232 has been reviewed and is approved for publication.

APPROVED: Edward J. Cupples

EDWARD J. CUPPLES
Project Engineer

APPROVED:

ROSS H. ROGERS, Colonel, USAF
Chief, Intelligence & Reconnaissance Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (IRAA), Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

DISCLAIMER NOTICE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

19

1. REPORT NUMBER: RADC-TR-78-232 (18)

2. GOVT ACCESSION NO.:

3. RECIPIENT'S CATALOG NUMBER:

4. TITLE (and Subtitle): (6) THE SPEECH ENHANCEMENT ADVANCED DEVELOPMENT MODEL,

5. TYPE OF REPORT & PERIOD COVERED: (9) Final Technical Report, Oct[redacted] 75 - Feb[redacted] 78

6. PERFORMING ORG. REPORT NUMBER: N/A

7. AUTHOR(s): (10) Mark R. Weiss, Ernest Aschkenasy

8. CONTRACT OR GRANT NUMBER(s): (15) F30602-76-C-0033 New

9. PERFORMING ORGANIZATION NAME AND ADDRESS:
Depa[redacted]t. of Computer Sciences New
Queens College
Flushing NY 11367

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS:
~~31011G~~
~~20330301~~

11. CONTROLLING OFFICE NAME AND ADDRESS:
Rome Air Development Center (IRAA)
Griffiss AFB NY 13441

12. REPORT DATE: (11) Nov[redacted] 78

13. NUMBER OF PAGES: 122

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Same

(12) 127 p.

15. SECURITY CLASS. (of this report): UNCLASSIFIED

15a. DECLASSIFICATION DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer: Edward J. Cupples (IRAA)

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Speech enhancement
Narrowband noise suppression
Impulse noise suppression
Audio signals

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This report describes the design, principles of operation, and performance characteristics of an Advanced Development Model of a speech enhancement unit. This unit improves the quality and intelligibility of speech signals by the removal of frequently encountered interference or noise from received or recorded speech signals. A high speed digital array processor and various time and frequency domain algorithms permits the detection and attenuation of narrowband noise (such as tones, hums, whistles, etc.) and impulse noise (such as ignition pulses, static, etc.) with minimum degradation to the speech signals.

DD FORM 1 JAN 73 1473

UNCLASSIFIED 410 994

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

79 01 02 002

The enhancement unit provides automatic tracking and attenuation of interferring signals in real time and with a maximum lag of .15 second.

The heart of the speech enhancement unit is a powerful computer known as a macro-array processor, or MAP, that performs all of the measurement, analysis, and processing of the input signal. It is supported by a digital magnetic tape unit used to program the MAP and a minicomputer which reads the program into the MAP.

Tests on the unit showed attenuation of 30 to 50 db on both narrowband and impulse noise. Operational tests performed by trained Air Force personnel showed the unit to be highly effective in providing improved intelligibility and listenability which significantly reduced listener fatigue.

Provision has been made in the design and fabrication of the speech enhancement unit to implement a technique for attenuating wideband random noise. This technique known as INTEL is one of the few known methods of suppressing this commonly encountered noise without severely distorting co-existing speech.

The unit housed in a standard 19 inch rack, 52 inches high, and is powered by 110 volts, 60 Hertz power.

CONTENTS

ILLUSTRATIONS

Figure

ILLUSTRATIONS (continued)

Figures

TABLES

EVALUATION

This effort is part of the Center program being conducted under Project 2033 to improve the quality and intelligibility of speech signals that are masked and interferred with by impulse and/or narrowband noise.

Under this effort, an Advanced Development Model of a Speech Enhancement unit was designed and fabricated. This unit, which uses a high speed digital array processor and various time and frequency algorithms, provides an on-line real-time capability to remove frequently encountered communication channel interferences such as tones, buzzes, pops, hiss, etc., with minimum degradation to the speech signals. Its operations are fully automatic and provides the capability to automatically track and attenuate the interferring signals. Tests showed the unit to provide 30 to 50 db attenuation to narrowband and impulse noise. Operational tests performed by trained Air Force personnel showed the unit to be highly effective in providing improved intelligibility and listenability which significantly reduced listener fatigue.

Presently the speech enhancement unit is being modified to incorporate a technique for attenuating wideband random noise. This technique, known as INTEL, is one of the few known methods of suppressing this commonly encountered noise without severely distorting co-existing speech.

The techniques and methods developed under this effort will have significant impact on speech technology since most speech systems as word recognition, speaker identification, etc., require good quality signals in order to provide effective results. The speech enhancement unit can do much to improve the quality of degraded speech signals to an acceptable level for these systems.

Edward J. Cupples

EDWARD J. CUPPLES
Project Engineer

1.0 INTRODUCTION

The quality and intelligibility of speech that is transmitted over telephone-bandwidth communication systems frequently is degraded at the receiver by noise that is received along with the speech. In many installations that use speech communication systems, trained personnel often spend hours listening to received transmissions, sorting speech from non-speech signals, and trying to extract information from speech signals that have been obscured by noise. When the quality of the received signal is particularly poor, the monitor of a communication channel may have to hear a message several times, either by having it retransmitted or by playing back a recording of it, before he can understand it fully. The need for a means of attenuating communication channel noise has existed for a long time. At some Air Force installations, it has been a standing joke to direct a new, inexperienced monitor to requisition a "QRM" filter when he is bothered by impulse or tonal noises. The system that is described in this report is just such a device.

A large number of different types of noises are heard on speech communication channels. The most commonly occurring class of noises includes impulses, static, tones, buzzes, and hiss (wideband random noise). These non-speech like sounds affect speech intelligibility by masking fragments of speech sounds in time or components of speech sounds in frequency. To enhance speech that has been obscured by noise, it is necessary to remove the masking sound or, if possible, reduce its effect, without at the same time attenuating or distorting the obscured speech sounds.

Equipment and devices for enhancing the intelligibility of speech have been developed over many years. Among the more commonly used techniques are peak clipping to attenuate impulses, notch or comb filtering to attenuate constant frequency tones, and high and low pass filtering to attenuate wideband noise. However, these techniques, and others like them, are inadequate for the needs of a high-performance general purpose speech enhancer. Consider for example, the use of notch filters to attenuate tones. If the frequencies at which the tones will occur are known in advance, the filters can be pretuned to those frequencies and left on at all times or turned on whenever the tones are detected. If the frequencies are not known or are variable, the tuning of the filters can be controlled by phase-locked loops. However, this approach becomes extremely complicated if it is desired to track tones that are variable in number and that change rapidly and erratically in frequency. Moreover, if the amplitudes of the tones are not significantly larger than those of the components of speech this method can be ineffective, or worse, it can lead to the attenuation of speech components. In a similar way, peak clipping becomes an increasingly unsatisfactory method of attenuating impulse noise as the average rate of occurrence of impulses increases. When impulses are relatively infrequent, peak clipping is capable of providing adequate attenuation of the average impulse noise power. However, peak clipping leaves intact that fraction of an impulse that falls below the clipping level. Consequently, when impulses occur frequently, (for example, when they are generated by ignition noise), the power in these residual fractions can still degrade the quality and the intelligibility of the speech signal.

The long term objective of the work performed under this contract was to develop and implement a speech enhancer that would be fully automatic, operate in real time, and attenuate all of the common types of communication channel noise. At the time work began, some of the needed signal enhancement techniques had been simulated in simple form and had been shown to be effective, but required some additional research and development. Others were still in the early stages of their development. Still others were being considered for possible future development. To make it possible for the speech enhancement equipment to ultimately incorporate all current and future signal processing techniques, it was designed to be capable of simple modular expansion. The immediate objective of this contract was to accomplish this design and to complete and implement techniques for detecting and attenuating two types of noise--tones and impulses. Implementation of more sophisticated techniques, such as the INTEL method for attenuating wideband random noise, would await the results of tests of the system. These tests were to be made at an Air Force site over an extended period of time under realistic and practical conditions.

One of the implemented techniques is based on a unique system for attenuating tonal noises that was developed by the authors in 1968. Known as coherent spectrum shaping, it used a real-time spectrum analyzer, known as a coherent memory filter to transform a time-domain signal to and from the frequency domain. By inspecting a display of the spectrum, and experienced speech analyst could distinguish components of tonal noise from those of speech. The noise components could be attenuated by setting to zero those regions in the signal spectrum where the energy of the tones was concentrated, and then transforming the signal back to the time domain. Each region to be attenuated had to be specified by manually setting controls that determined the width and location of the region. Thus, both identification and attenuation of tonal components of noise required manual procedures. As in the system described above, the speech enhancer developed under this contract implements the concept of transforming a signal to the frequency domain, modifying it, and then retransforming it to the time domain. However, unlike the earlier system, it also implements fully automatic procedures for detecting and attenuating components of tonal noise. These are supplemented by manual procedures, similar to those described above, that can be used when needed.

The speech enhancer also implements a technique, developed under this contract, for detecting and attenuating noise impulses in the time waveform of the signal. Unlike peak clipping, this method removes almost all of the energy of the impulses. Moreover, it "smooths" the time waveform in the region of a removed impulse in such a way as to eliminate residual transients and discontinuities.

Not implemented in the speech enhancer, but bearing heavily on its general design, is a technique for attenuating wideband random noise. This procedure,which is known as INTEL, is one of the few known methods of suppressing this type of noise without at the same time severely distorting coexisting speech. Present plans are to incorporate the INTEL process in the speech enhancer during a later phase of its development.

During the development of the signal processing techniques that are implemented in the speech enhancer we made extensive use of a Sigma 7 computer and an associated real-time data aquisition system to study and test various detection and attenuation schemes. The input signals for these studies were tape recordings of speech, recordings of several kinds of noise, and recordings of speech plus noise at S/N down to -30dB. The data aquisition system sampled the recorded data at a 10-kHz rate with 16-bit accuracy, converted the sampled values to binary form, and recorded them onto digital computer tape.

Two sets of audio recordings were used for the studies and tests. In the first set, speech selections were obtained from the recorded speech of four adult talkers, three male and one female. Tonal noise was obtained from several signal generators and included single tones, harmonic sequences, and randomly spaced sequences containing up to ten tones at varying amplitudes. Their durations ranged from 30 seconds to 50 ms; their stabilities ranged from high (i.e., constant in frequency and amplitude) to very low (e.g., tone frequencies changing at rates up to 10,000 Hz per second). A second set of audio test data recordings were obtained from transmissions of speech over noisy communication channels. These recordings, representing as they did a realistic and practical conditions, were used to test and evaluate the impulse and tone attenuation processes as their developments neared completion and provided a means of "fine tuning" these processes.

The contract under which this work was performed refers to the speech enhancer that is described in this report as an Advanced Development Model (ADM), which it defines as: An item used for experimentation or tests to (a) demonstrate the technical feasibility of a design, (b) determine its ability to meet existing performance requirement, (c) secure engineering data for use in further development and where appropriate, (d) establish the technical requirements for contract definition. In view of the experimental objectives of the ADM, the system was designed to provide a wide range of features and adjustments whose usefulness and effectiveness could be tested. The tests and evaluations planned for the speech enhancer ADM were performed at an Air Force test facility.

The system that was completed and delivered to the Air Force for testing is shown in figure 1. It is composed of five units, housed in a 52-inch high cabinet rack. The heart of the ADM is a very powerful computer, known as a macro-arithmetic processor, or MAP, that performs all of the measurements, analyses, and processing of input signals. It is supported by a number of peripheral units, as shown schematically in figure 2. The programs that control the MAP are stored permanently on digital magnetic tape. When operation of the ADM is initiated, a minicomputer reads this tape and transfers the program to the memory of the MAP. Analog input signals are converted to digital form by the input unit, and processed, first to attenuate tonal noises and then to attenuate impulse noises. The enhanced signals are then adjusted in level, fed out of the MAP (in digital form), and converted by the output unit to analog form. Selected analysis data from the MAP also are converted to analog form, by the display unit, and displayed on an oscilloscope. The important characteristics and performance features of the ADM are listed in Table 1.

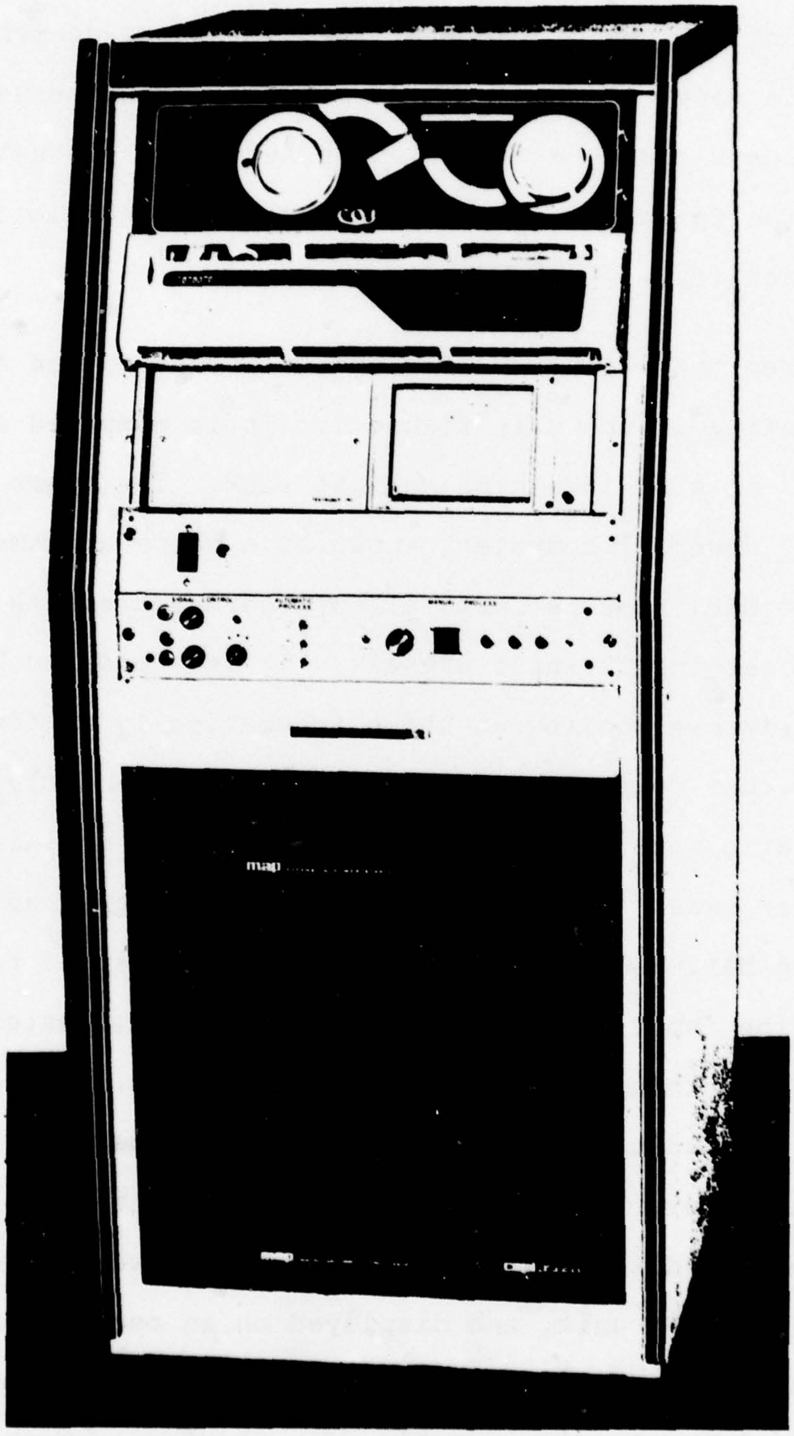


FIGURE 1 PHOTOGRAPH OF THE SPEECH ENHANCEMENT ADM

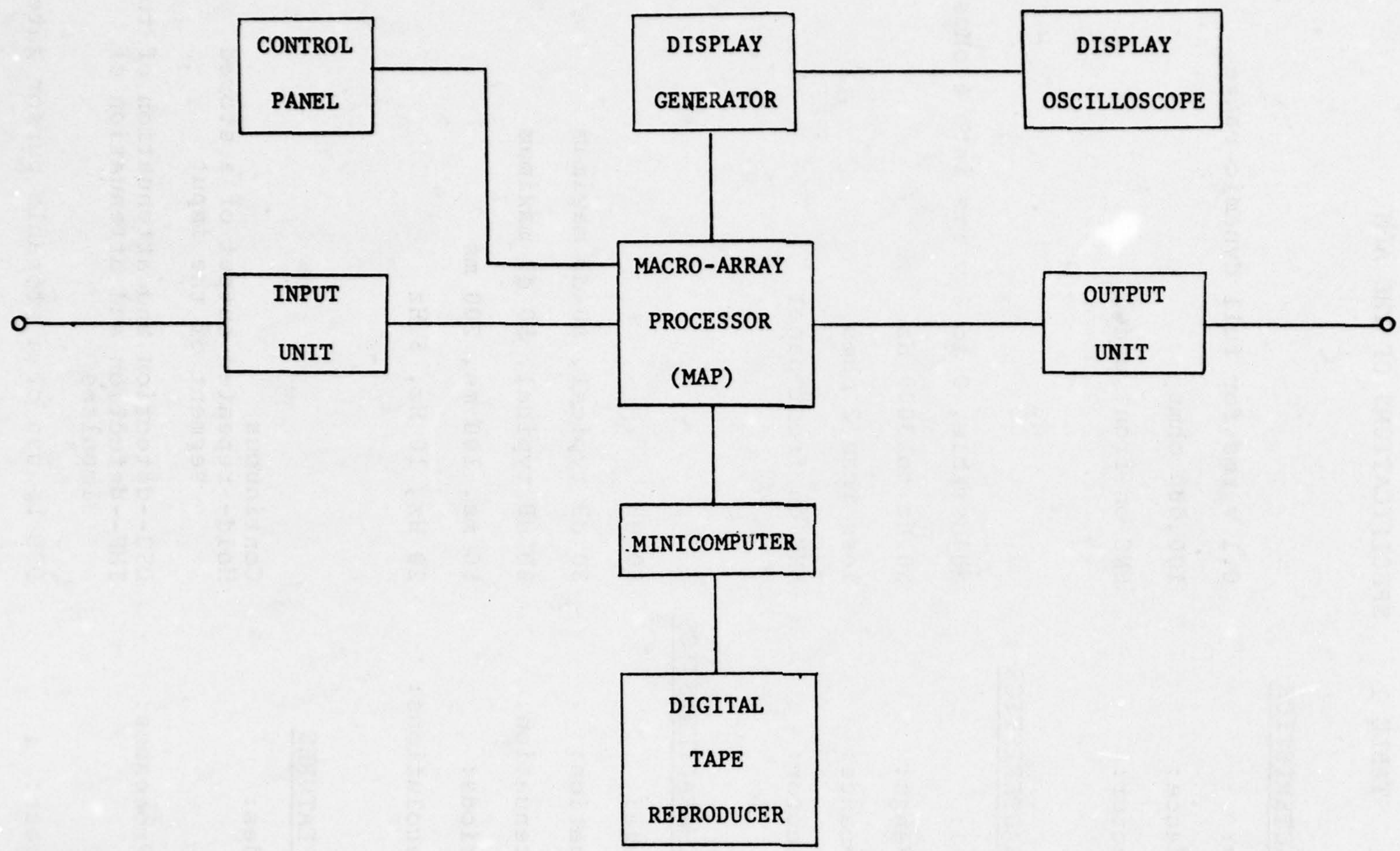


FIGURE 2 FUNCTIONAL DIAGRAM OF THE ADM

TABLE 1 SPECIFICATIONS OF THE ADM

INPUT CHARACTERISTICS	
Sensitivity:	0.1 v rms for full dynamic range
Input Impedance:	100,000 ohms
Input Connector:	BNC on front panel
OUTPUT CHARACTERISTICS	
Signal Level:	adjustable, 0 to 3 v rms into 8 ohms
Frequency Range:	50 Hz to 3000 Hz
Output Impedance:	less than 2 ohms
Output Connector:	BNC on front panel
PROCESSOR CHARACTERISTICS	
Dynamic Range:	60 dB
Tone Attenuation:	30 dB typical, 50 dB maximum
Impulse Attenuation:	30 dB typical, 50 dB maximum
Process Periods:	50 ms, 100 ms, 200 ms
Spectrum Resolutions:	20 Hz, 10 Hz, 5 Hz
PROCESSOR FEATURES	
Process Modes:	Continuous Hold--repeated output of a stored segment of the input
Automatic Processes:	DSS--detection and attenuation of tones IMP--detection and attenuation of impulses
Manual Process:	DSS by use of a storable cursor gate

TABLE 1 (Continued)

Cursor Gate Features

Modes:	Single, Harmonic
Location Range:	DC to 3000 Hz
Width Range:	15 Hz to 105 Hz (process period=200 ms)
	30 Hz to 210 Hz (process period=100 ms)
	60 Hz to 420 Hz (process period= 50 ms)

DISPLAY CHARACTERISTICS

Functions Displayed:	(1) Amplitude Spectrum
	(2) Cursor Gate (manual DSS process)
	(3) Stored Gates (manual DSS process)
	(4) Attenuated Regions of the Spectrum
	(5) Input Signal Level
Refresh Rate:	80 displays per second

MISCELLANEOUS CHARACTERISTICS

Power Required:	110 volts, 17 amperes, 60 Hz
Dimensions:	Height 52 inches
	Width 22 inches
	Depth 32 inches
Weight:	550 pounds

2.0 THE TONE ATTENUATION PROCESS

Essential to any technique for the reduction of noise is the ability to separate the signal from the noise to the greatest extent possible. Once isolated, the noise can be discarded or, at least attenuated. In the impulse attenuation procedure, which is described in Section 3, the impulses occupy narrow regions of the time waveform of the speech signal, making them easy to isolate from most of the signal. Sustained noises, such as tones or wideband random noise, coexist with the signal and cannot readily be separated from it by simple inspection of the time waveform. However, by transforming the signal to the complex frequency domain, the components of tonal interference can be identified in the amplitude spectrum. Once identified, tones can be attenuated by reducing their amplitudes in the complex spectrum and then transforming the spectrum back to the time domain.

The procedure described above consists of four steps:

1. Time-to-frequency transformation of the input signal;
2. Detection of components of tonal noise;
3. Attenuation of detected tones;
4. Frequency-to time transformation of the signal;

Each step is explained and the procedures involved described in the sections that follow. The computer programs that implement the procedures are described in Section 5, and the associated hardware in Section 4.

2.1 Time-to-Frequency Transformation.

The fast Fourier transform (FFT) procedure is used to compute short term complex spectrums of the input signal. In preparation for entry into the computer, the input signal is sampled at a 10-kHz rate and the samples converted to 14-bit binary numbers. Segments of the signal 51.2 ms, 102.4 ms, or 204.8 ms in duration are amplitude-weighted by a symmetrical triangular waveform. These intervals, which are referred to as analysis windows and also as process periods, contain respectively 512, 1024, or 2048 successive samples of the input signal. Correspondingly, the spectrum of the signal within each window is determined at intervals of 19.53 Hz, 9.77 Hz, or 4.88 Hz. For convenience, nominal values will be used in the descriptions that follow. Thus, the process period durations are 50 ms, 100 ms, and 200 ms, and the corresponding spectrum sample spacings are 20 Hz, 10 Hz, and 5 Hz.

To permit analysis and processing of the complete signal, the analysis window is moved in half-window steps, as illustrated in figure 3. Thus, each half-window segment is processed twice, first as the upper half of a window and then as the lower half of the succeeding window. As is described in Section 2.4, the 50 percent overlapping of successive windows makes it possible to reconstruct a continuous, unweighted output signal.

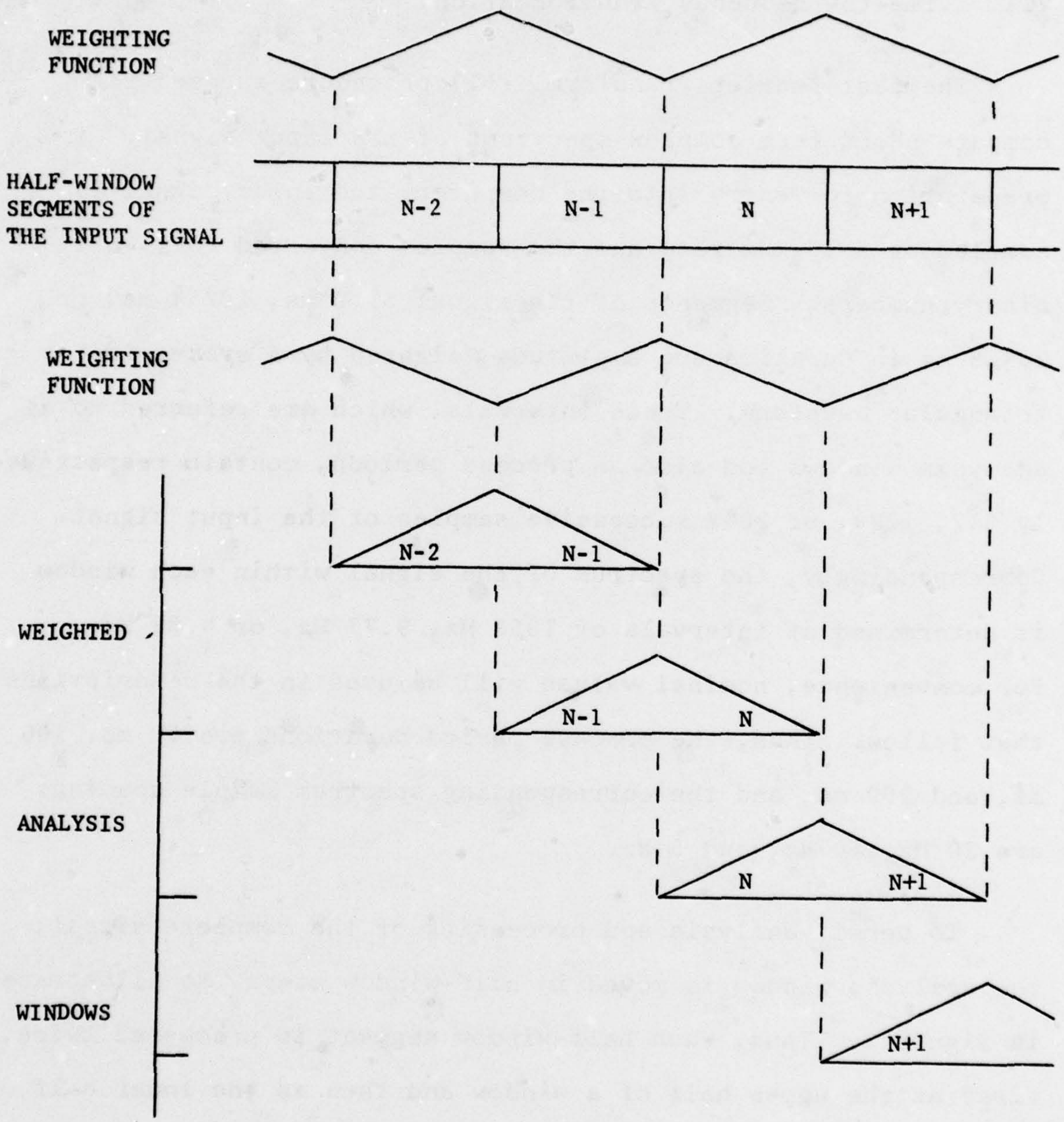


FIGURE 3 OVERLAP WEIGHTING OF THE INPUT SIGNAL

Since the signal is sampled at a 10-kHz rate, the complex spectrum calculated by the FFT will extend to 5 kHz. But, since the range of interest extends only to 3 kHz, the complex spectrum samples above that frequency are set to zero. The samples below 3 kHz are stored,and they also are converted to absolute form to generate an amplitude spectrum. The amplitude spectrum is used by the automatic processes, described below, for detection of components of tonal noise. They also are used in the manual detection process, as described in Section 2.4.

2.2 Automatic Detection of Components of Tonal Noise

Some amplitude spectrums of speech, of tonal noise, and of combined speech and noise are shown in figure 4. It is evident from waveshapes such as these that the peaks that correspond to tones cannot be identified merely from their shapes, amplitudes, and locations in the spectrum. Other criteria must be used, criteria based on measurable characteristics in which the spectrum peaks of speech and of noise differ significantly and consistently. The two measurable characteristics that are most useful are (1) the stabilities of the two classes of sounds, and (2) the amplitude distribution of their significant components.

2.2.1 Stability Criteria and the Peak Matching Procedure

Speech is a continuously varying sound, with components that seldom are constant in frequency or amplitude for more than

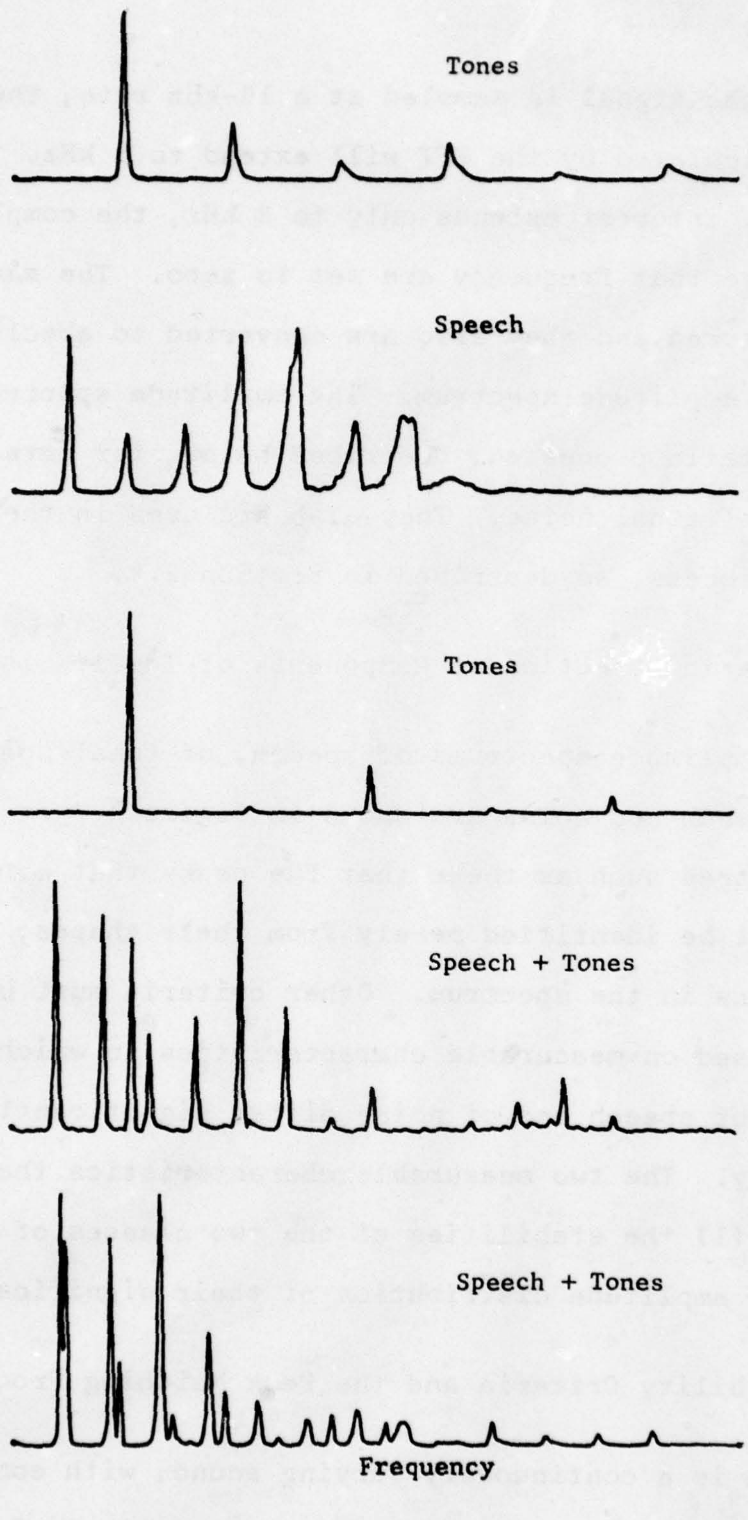


FIGURE 4 AMPLITUDE SPECTRUMS OF SPEECH, TONES, AND SPEECH PLUS TONES

25 msec. Tonally structured noises, on the other hand, encompass a wide range of stabilities. Fortunately, most of the noises of this type that are encountered in practical situations are more stable than speech, with components essentially constant for periods greater than 50 ms. To detect the components of such noises, the automatic DSS process examines each amplitude spectrum to identify peaks that are stable. It does this by comparing the location and amplitude of each peak with the locations and amplitudes of peaks on the preceding amplitude spectrum. Peaks on the two spectrums that match in location and amplitude within specified limits are identified as being stable and therefore as representing components of noise. This condition is illustrated in figure 5, with relatable peaks on adjacent amplitude spectrums linked by dashed lines.

The limits within which the peaks of "stable" tones must correspond to constitute a match must be wide enough to accommodate short-term instabilities but no so wide as to match either unrelated peaks or slowly varying speech harmonics. The optimum limits, shown in Table 2, were determined by testing various combinations of limits on over 3000 spectra of speech, tones, and speech plus tones.

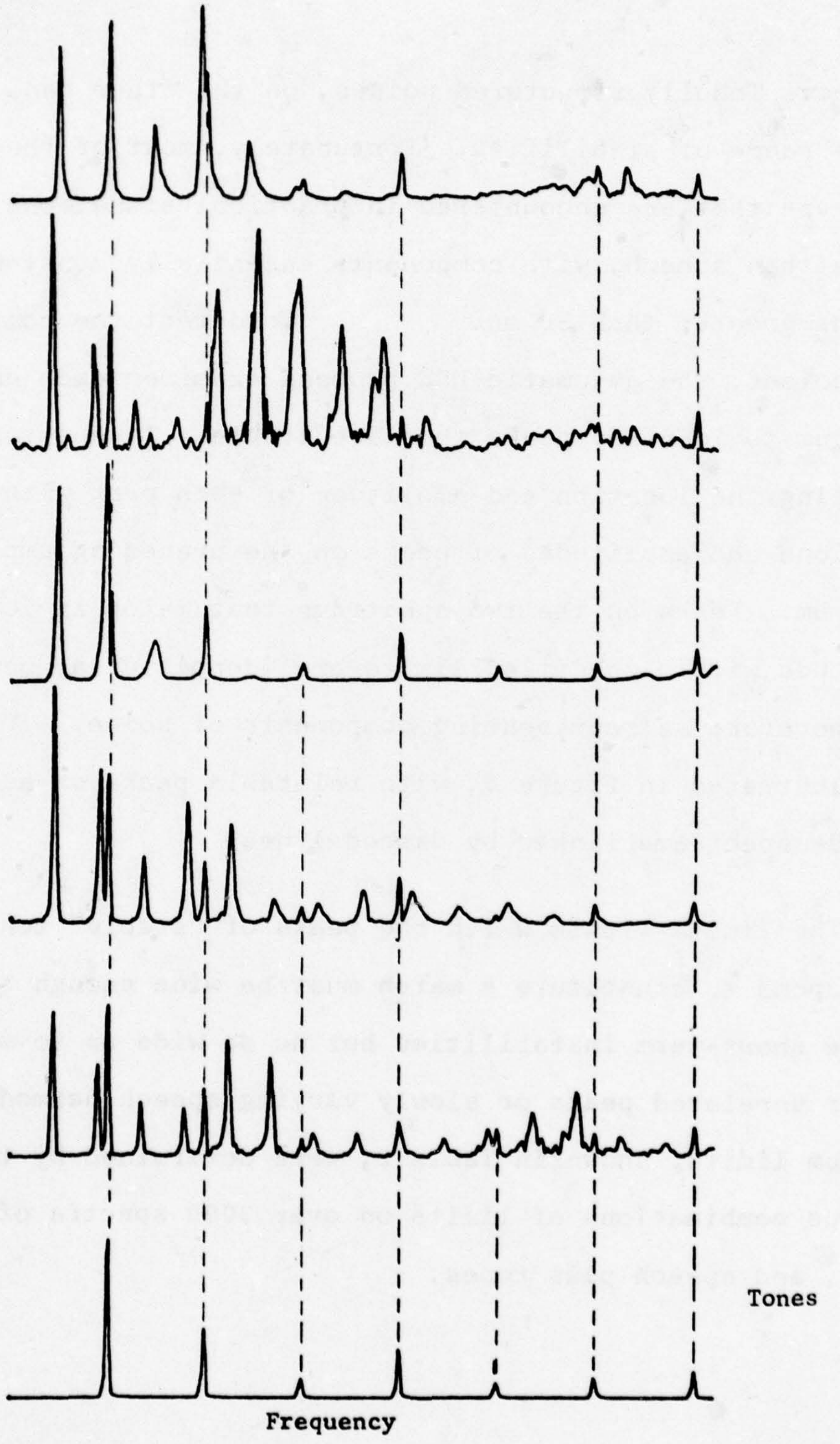


FIGURE 5 DETECTION OF TONES BY MATCHING OF SPECTRUM PEAKS

TABLE 2

Process Period	Sample Spacing	Frequency Limits	Amplitude Limits
(ms)	(Hz)	(Hz)	(dB)
50	20	40	0.6
100	10	20	1.2
200	5	10	2.4

In general, the peak matching procedure is most effective for the 100 ms and 200 ms process periods, for which it will correctly match the peaks of tones whose frequencies change at rates up to 480 and 240 Hz per second respectively. Although the 50 ms process period provides the ability to match peaks that move even faster, it can lead to an unacceptibly large number of speech harmonics being matched when pitch is varying slowly.

The amplitude limits increase with increasing process period, and the frequency limits decrease. The increasing amplitude limits reflect the need to allow for the larger changes in amplitudes of tones that are observed as the process period is increased. The decreasing frequency limits reflect the need to use the smallest practical frequency deviations (± 2 spectrum samples) in order to minimize the likelihood of matching peaks that represent speech harmonics.

2.2.2. Amplitude Criteria and the Detection Threshold Procedure

The spectrums of voiced speech sounds typically exhibit a large number of equally spaced peaks. By contrast, most tonal noises (with the exception of low-pitch buzzes) usually contain far fewer components than speech and so their spectrums exhibit far fewer peaks. Consequently, the ratio of the largest peak in a spectrum to the average level of the spectrum is likely to be lower for speech than it is for noise. This characteristic is the basis of the second method for automatically discriminating tone peaks from speech peaks in the spectrums of the input signal. The technique, as illustrated in figure 6, is to compute a detection threshold as some factor times the average spectrum level, and to identify as tones those peaks that exceed the threshold.

During the development of this procedure we tested a number of ways of computing the detection threshold. Among these were multiples of the rms spectrum level, the mean-squared spectrum level, and the average and peak levels of the input time waveform. We found that detection thresholds that were based on the average spectrum level were the most effective ones for discriminating tone peaks from speech peaks and were the least dependent on the speech characteristics of different talkers.

The choice of an optimum factor for multiplying the average spectrum level to compute the detection threshold was based on an analysis of the speech of four talkers (three male and one

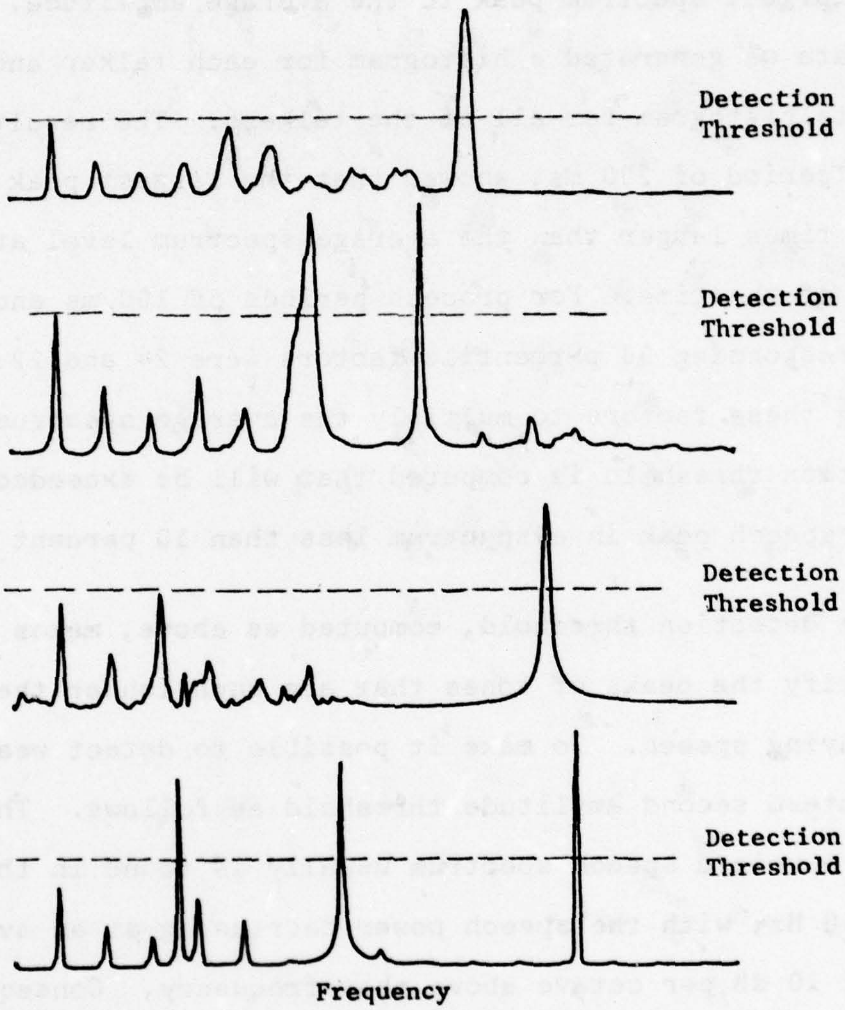


FIGURE 6 DETECTION OF TONES BY USE OF AN AMPLITUDE THRESHOLD

female). We examined spectra for 30 seconds of speech for each talker. For each spectrum we computed the ratio of the amplitude of the largest spectrum peak to the average amplitude. From these data we generated a histogram for each talker and then a composite histogram for all of the talkers. The results, for a process period of 200 ms, showed that the largest peak was more than 28 times larger than the average spectrum level at least 90 percent of the time. For process periods of 100 ms and 50 ms the corresponding 90 percentile factors were 24 and 22. Hence, by using these factors to multiply the average spectrum level, a detection threshold is computed that will be exceeded by the largest speech peak in a spectrum less than 10 percent of the time.

The detection threshold, computed as above, makes it possible to identify the peaks of tones that are much louder then those of accompanying speech. To make it possible to detect weaker tones, we compute a second amplitude threshold as follows. The largest peak in a voiced speech spectrum usually is found in the region below 750 Hz, with the speech power decreasing at an average rate of about 10 dB per octave above that frequency, Consequently, the threshold as computed above is increasingly unlikely to be exceeded by speech peaks for frequencies above 750 Hz. Therefore, the threshold can be made more sensitive by decreasing its amplitude with increasing spectrum frequency. One way to do this is to force the threshold to follow the average speech spectrum power curve described above. However, this characteristic is subject to talker-to-talker variations, and assumes the use of a high quality microphone and a flat transmission characteristic in the communication channel. A much more reliable procedure is to divide the spectrum into four bands, each 750 Hz wide, and compute an optimum threshold for each band. By performing histogram analyses similar to that described earlier, we determined the 90 percentile ratios of largest spectrum peak to average spectrum level for each band and for each process period. These are shown in Table 3.

TABLE 3

OPTIMUM FACTORS FOR COMPUTING THE DETECTION THRESHOLD IN 750-Hz WIDE BANDS

	Band			
Process Period (ms)	1 50 - 750 Hz	2 750 - 1500 Hz	3 1500 - 2250 Hz	4 2250 - 3000 Hz
50	7.5	5.0	4.6	4.0
100	9.5	5.4	4.9	4.0
200	10.1	5.9	4.9	4.6

It would seem reasonable that the detection procedure needs to use only these thresholds since they would be exceeded by any peak that exceeds a threshold that is 28 times the average level of the full spectrum. However, in some circumstances, this may not be so. For example, if three or more large peaks occur in band 2, 3, or 4, their contribution to the average level in that band could cause the calculated threshold to be above the peaks. Therefore, to increase the chances of detecting tonal peaks wherever they occur, the detection procedure is performed twice, the first time using a single threshold calculated across the entire spectrum and the second time using the four thresholds calculated in the four bands.

2.3 Attenuation of Detected Tones

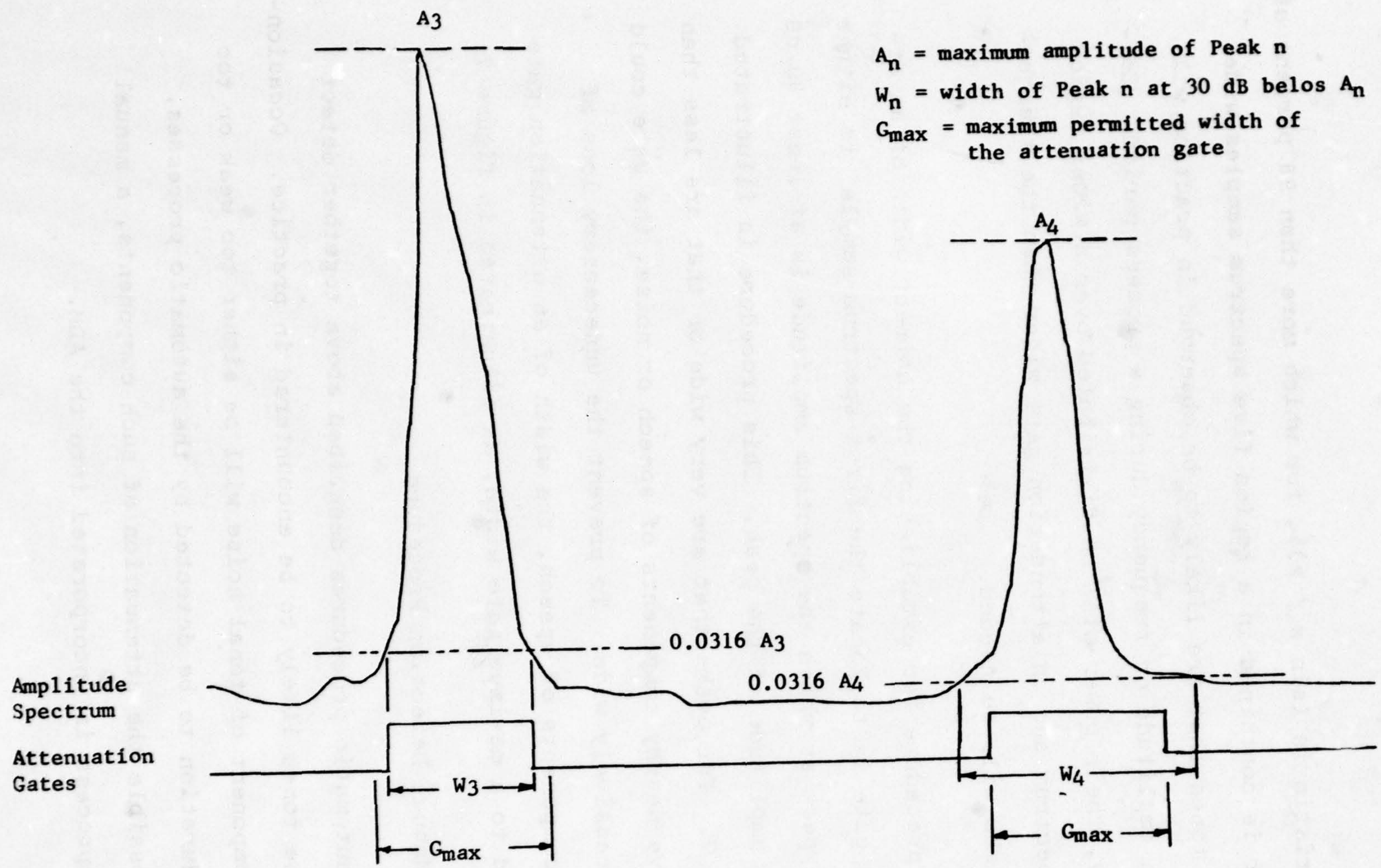
Tones whose peaks are detected in the amplitude spectrum are attenuated 40 dB in the corresponding complex spectrum. This is accomplished by multiplying by 0.01 those components of the complex spectrum that fall in a narrow region centered about each detected peak. For peaks detected by the matching procedure the minimum width of the attenuation region (which also is referred to as an attenuation gate) is three spectrum samples wide. That is, it includes the sample at which a peak is located and the first sample to either side of the peak. For peaks detected by the detection threshold, the minimum attenuation gate includes all the samples in the region in which a peak exceeds the threshold and in addition the first sample to either side of that region.

The total power in a tone is not concentrated at the center of its peak in the spectrum but is distributed in accordance with the spectrum selectivity characteristic. For triangularly weighted absolutely stable signals, the amplitude selectivity characteristic is $(\sin x / x)^2$, for which more than 98 percent of the power is contained in a region five spectrum samples wide. However, tones that are likely to be observed in practice will change in amplitude or frequency during a process period. Consequently, their power will be distributed over a wider region in the spectrum and an attenuation gate wider than the theoretical minimum will be needed.

The procedure for establishing the correct width of an attenuation gate is to locate the first spectrum sample to either side of a peak at which the spectrum amplitude is at least 30 dB below the amplitude of the peak. This procedure is illustrated in figure 7. For peaks that are very wide or that are less than 30 dB above nearby components of speech or noise, the gate could become excessively wide. To prevent the unnecessary loss of spectrum components of speech, the width of an attenuation gate is limited to a maximum gate width, as illustrated in figure 7.

2.4 The Manual Detection Procedure

The automatic procedures described above together detect most of the tones likely to be encountered in practice. Occasionally, a component of tonal noise will be either too weak or too brief in duration to be detected by the automatic processes. To make possible the attenuation of such components, a manual detection process is incorporated into the ADM.



A_n = maximum amplitude of Peak n
W_n = width of Peak n at 30 dB belos A_n
G_{max} = maximum permitted width of the attenuation gate

FIGURE 7 AUTOMATIC ADJUSTMENT OF ATTENUATION GATE-WIDTH

The manual process consists of two systems, the display and the data entry controls. Amplitude spectra are displayed, as shown in figure 8. On the trace below the spectrum is shown the indicator of an attenuation gate whose location and width can be adjusted by use of manual controls. This indicator, which we refer to as the cursor, points to the region of the spectrum that will be attenuated by the gate it represents.

In using the manual system, the operator examines the spectrum display while listening to the output signal. An intermittent component usually can be located by correlating the appearance of a peak in the spectrum display with its occurence in the audio output signal. Continuous tones that are too weak to be located visually can be detected by moving the cursor through the spectrum until the tone disappears in the output. Tones that are both weak and intermittent still can be located by use of a control that "freezes" a segment of the input signal. Normally, successive process-period long segments of the input signal are analyzed in the order in which they arrive. By use of this control, the segment currently being analyzed is not replaced by the succeeding one but is stored and reprocessed repeatedly. By storing a segment that contains a weak, intermittent tone the operator can hear it repeatedly while examining the spectrum or adjusting the location of the cursor.

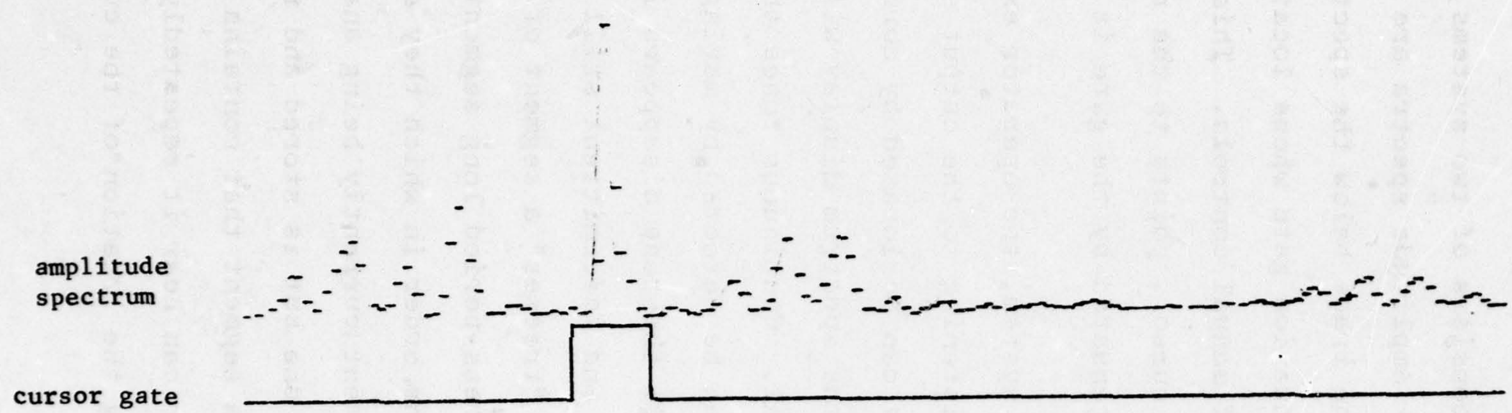


FIGURE 8 INFORMATION DISPLAYED FOR MANUAL SETTING OF ATTENUATION GATES

Once the cursor has been located as needed to attenuate a tone,the operator can store its location and width in the memory of the ADM. Attenuation of the tone will now be performed by the stored gate, leaving the cursor free to be turned off or, if need be, used to locate and attenuate a different tone. Means also are provided for deleting gates that have been stored in the memory when the tones they were intended to attenuate no longer are present in the input signal.

2.5 Frequency-to-Time Transformation

After all processing to attenuate tonal noise has been completed, the complex spectrum is transformed to the time domain by use of an inverse FFT algorithm. Tones that were attenuated in the spectrum will be correspondingly reduced in the regenerated time waveform. All that remains to do is to recombine overlapped analysis windows and to remove the triangular weighting that was imposed on the input signal. This is accomplished, as shown in figure 9, by adding the upper half of one analysis window to the lower half of the succeeding one. Since these represent the same half-window length segment of the input signal, with complementary weighting, combining them removes the weighting and restores the original envelope of the signal. The resulting output signal is continuous, with no discontinuities at the segment boundaries.

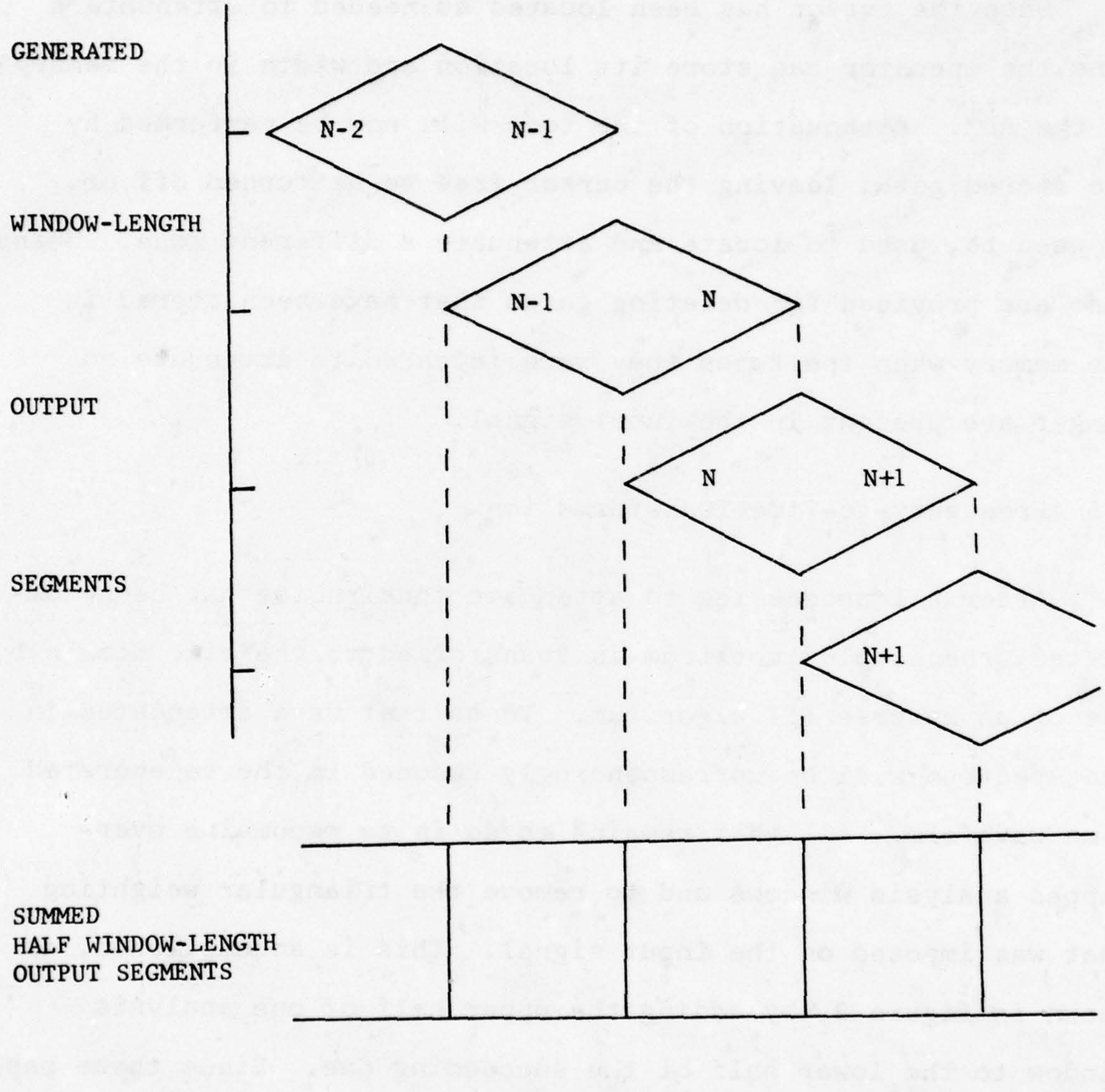


FIGURE 9 GENERATION OF AN UNWEIGHTED OUTPUT SIGNAL

3.0 IMPULSE ATTENUATION

Radio transmissions have long been plagued by bursts of noise that can last from several hundred microseconds to several seconds. Usually referred to as static, they arise from numerous causes. The most commonly thought of one, lightning, generates noise bursts that last from tens of milliseconds to seconds. A second common type of static is caused by brief arcs produced by high voltage breakdowns or by the separating of contacts that carry electrical current. This type of static is referred to as ignition noise when it is associated with the operation of the spark plugs and distributor in an internal combustion engine. It is called commutator or brush noise when it associated with electrical motors or generators. In either case, the noise bursts usually last less than 5 ms and occur in periodic or quasi-periodic patterns. A third kind of noise burst is caused by brief discharges in electrical components in radio receivers. In the absence of other, stronger noises, this "shot" noise can be a major source of interference for very weak signals.

Mathematically, an impulse is an event of infinitesimal duration. Colloquially, we can describe a noise burst that is short compared with speech sounds, as an impulse. More precisely, since phonemes in running speech average about 75 ms in duration, we can define noise bursts shorter than 25 ms as impulses. By this definition, impulses seldom mask entire phonemes.

Some typical waveforms of impulse are shown in figure 10.

Most impulses are less than 5 ms in duration. Consequently, impulses that carry enough power to obscure speech must be comparable to or greater than the speech peaks. For example, assume as a worst case that 5-ms long impulses occur at an average rate of 20 per second. Since their duty cycle is 10 percent, the peak amplitude of these impulses will be ten times the average amplitude and 3.16 times the rms amplitude. For speech, a widely used rule of thumb puts the speech peaks also at about 3 times the rms level. Therefore, at equal levels of power the impulses in this example will be about as high as the speech peaks. Obviously, the peak amplitude of narrower or less frequently occurring impulses (that is, for impulses with a shorter duty cycle) will exceed that of speech for equal power levels. This is illustrated in figure 11, which shows curves of the ratios of peak to rms level and peak to average level as a function of the average duty cycle of constant amplitude impulses. The technique of peak clipping as a means of attenuating impulses makes use of the characteristics illustrated in this figure. However, in peak clipping the clipping threshold usually is constant and set at a level above the largest speech peak likely to be observed. The method can be improved upon by making the clipping level vary with the input signal level.

FIGURE 10 IMPULSE WAVEFORMS OBSERVED IN RECEIVED RADIO SIGNALS

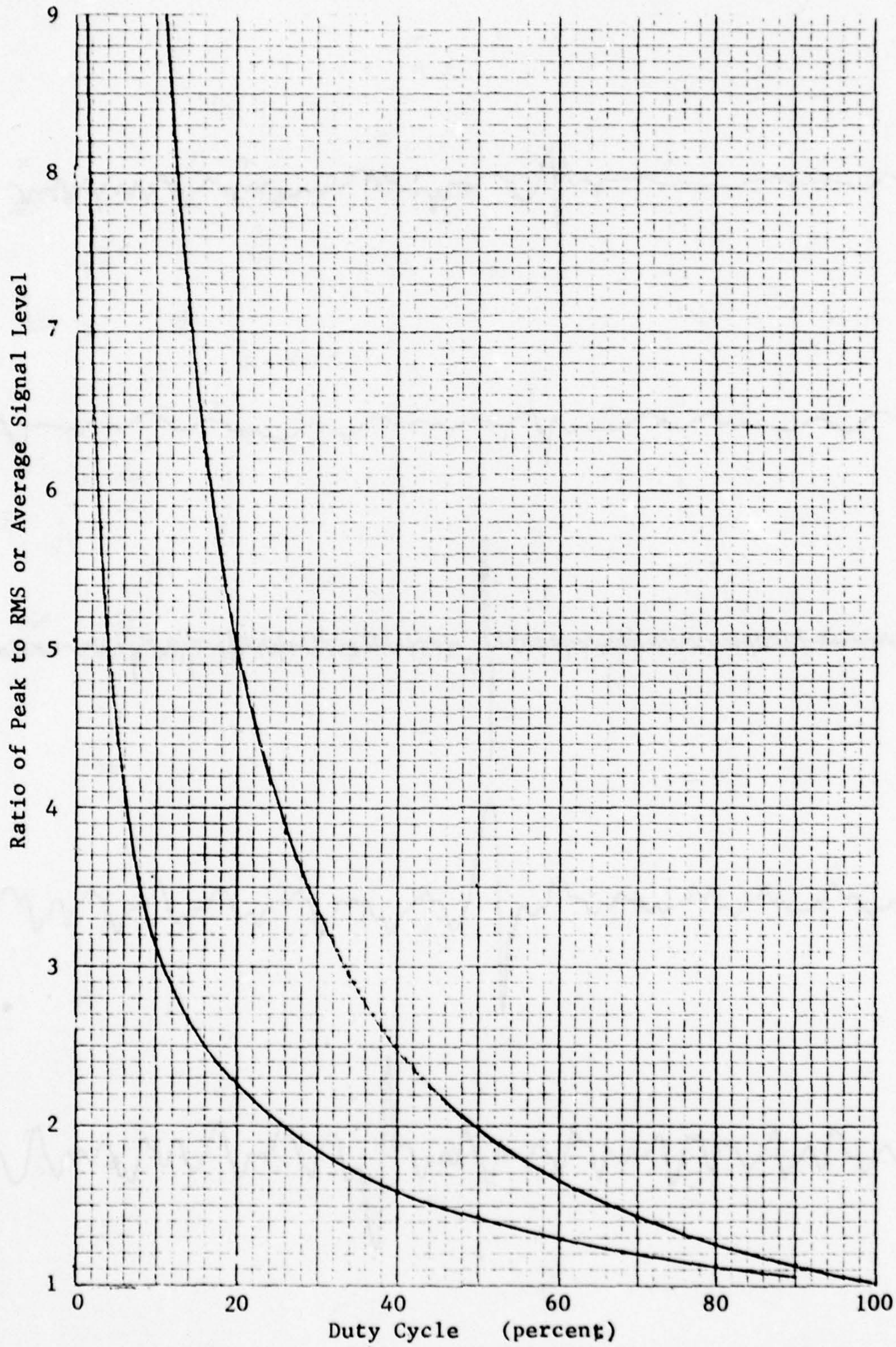


FIGURE 11 RATIOS OF PEAK LEVEL TO RMS LEVEL AND TO AVERAGE LEVEL FOR IMPULSES

A threshold such as that used for clipping impulses also can be used as a means of detecting them by identifying as impulses all peaks that exceed some multiple of the rms level of the signal. Using the rule of thumb cited above, by setting a "detection" threshold at four times the rms level the chances of identifying a speech peak as an impulse are very small. However, as shown in figure 11, such a threshold will detect impulses reliably only up to a duty cycle of 6 percent. This approach becomes much more useful if the detection threshold is based on a multiple of the average level. During our studies of speech and impulses, we computed the distribution of peak-to-average levels for the speech of four talkers. The results, shown in figure 12, show that fewer than 3 percent of speech peaks exceed five times the average level. From figure 11, it is seen that if the detection threshold is set at five times the average signal level, impulses can be distinguished from speech at duty cycles up to 20 percent.

The procedure described above discriminates effectively between speech and impulse signals. However, when both signals are present simultaneously, their average values will sum, and the computed detection threshold will be higher than it would be for impulses alone. Consequently, as the signal-to-noise ratio is increased from zero (i.e., no speech present),the maximum average duty cycle for which impulses will be detected will be

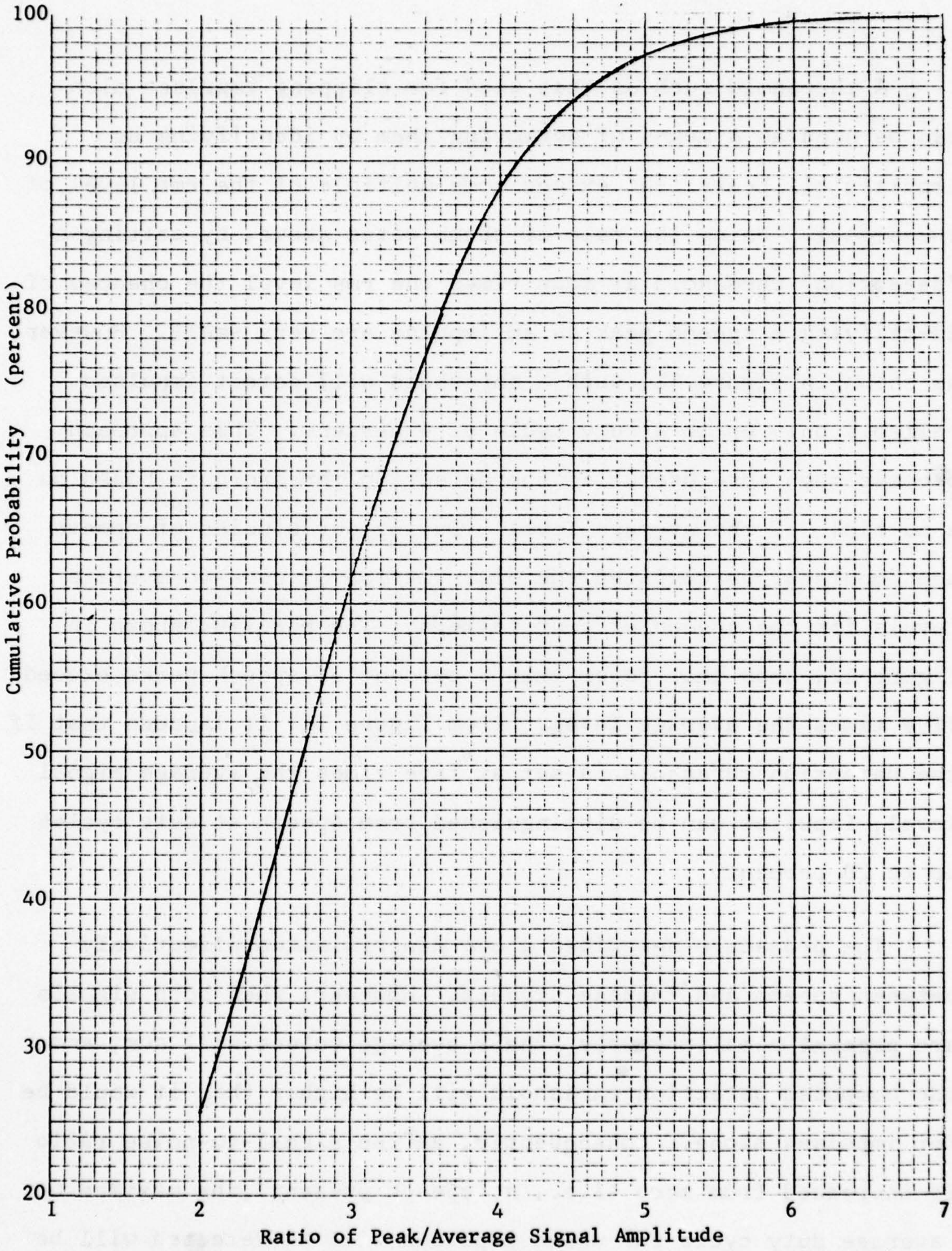


FIGURE 12 CUMULATIVE PROBABILITY OF RATIO OF PEAK TO AVERAGE LEVEL IN THE SPEECH OF FOUR TALKERS

reduced. This effect is shown in figure 13, for signal-to-noise ratios ranging from zero ($-\infty$dB) to 10 (+20dB).

Based on the data in figures 12 and 13, the impulse attenuation routine computes the detection threshold as five times the average signal level. At this threshold, 2.6 percent of speech peaks will be identified as impulses when only speech is present in the input signal. However, for most practical input signals, an adequate level of tonal, impulse, or random noise will be present to raise the detection threshold above the level of all of the speech peaks.

To remove a detected impulse from the time-waveform, the attenuation routine sets the waveform to zero at the location of the impulse. Before this can be done, it is necessary to define an attenuation zone at each impulse that includes as much of the impulse and as little of the adjacent speech as possible. Ideally, if the points at which the impulse begins and ends are known accurately, the entire impulse can be removed with no unnecessary loss of speech signal. In reality, it is impossible to determine the boundaries of impulse with adequate accuracy. Impulses vary greatly in waveshape, ranging from single, unipolar, steep-sided pulses to complex combinations of bipolar peaks with trailing ripples. Consequently, it can be difficult to tell where speech and impulses begin and end.

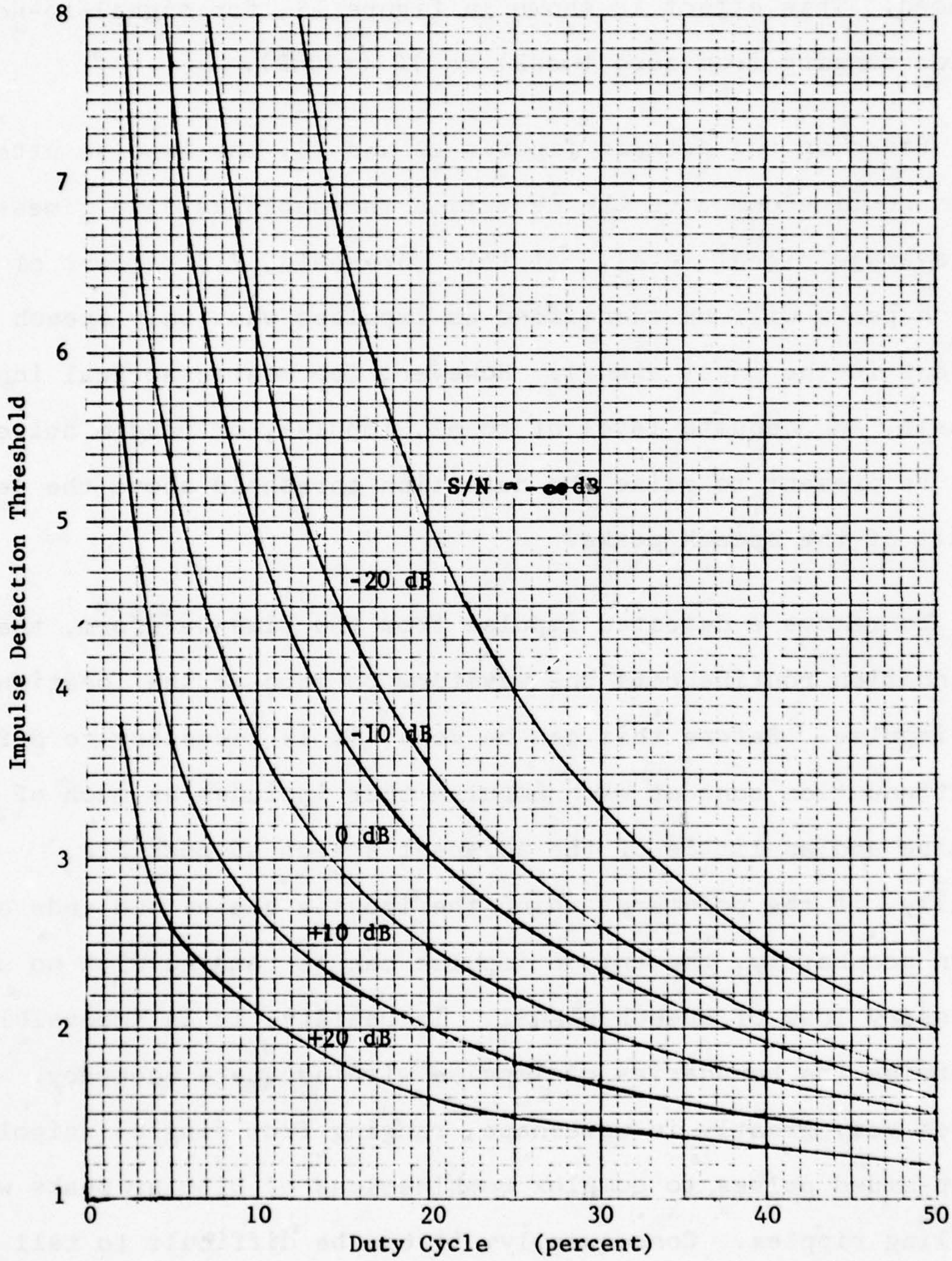


FIGURE 13 VARIATIONS OF DETECTION THRESHOLD WITH DUTY CYCLE AND S/N

Our solution to the problem of defining an impulse attenuation zone is to extend the region in which the impulse exceeds the detection threshold by adding guard bands at the detected leading and trailing edges. At the leading edge, the guard band must allow primarily for the time the impulse takes to rise to the level of the detection threshold. As described in Section 4.1, the input signal is limited to a band 3 kHz wide by a filter whose impulse response rise-time is about 200 microseconds. Thus, the required leading-edge guard band is 200 microseconds. The trailing edge guard band must allow for both the fall time of the impulse and the occurrence of ringing components generated by the input filter. Here, a band 700 microseconds wide was found to be adequate. Thus, the impulse attenuation zone is 900 microseconds wider than the detected width of the impulse.

If no further steps were taken in the processing of the signal,the attenuated regions of the time waveform would be perceived as gaps that frequently would be accompanied by strong clicks. The clicks would be caused by the abrupt changes in the sound level at the edges of the gaps. Of course, these transients can be muted by gradually decreasing the attenuation of the signal outside the impulse attenuation zones. But this will increase the amount of impulse-free speech signal that is lost. Moreover, if the impulses that are removed are closely spaced, the succession of gaps can all but destroy the intelligibility of the regenerated speech. Obviously, the impulse attenuation procedure requires one additional operation to eliminate these effects.

The solution to the problem created by the presence of gaps is to fill them. But if the "filler" itself is not to be perceived, it must preserve the continuity of the sounds perceived at either side of the gaps. Therefore, the filler material must be drawn from the time waveform in the vicinity of the gap. Following this approach, a segment of the waveform immediately preceding or following a gap could be replicated in the gap. While attractively simple, this procedure suffers from two weaknesses. First, it still leaves audible amplitude and/or phase transients at the edges of the gap. Second, it does not provide a smooth transition of sound between the filled gap and the non replicated region adjacent to the gap. However, the basic approach is sound and, with certain modifications, it becomes the needed third step in the impulse attenuation procedure.

The method that is used to fill a gap is illustrated in figure 14. As shown, the filler material is drawn from gap-length segments located at the leading and trailing edges of the gap. Each segment is amplitude weighted by a function that decreases linearly from unity at the edge of the gap to zero at the far end of the segment. The segments are then rotated about the gap edges so as to appear in the gap, and then they are summed. The filler that is produced by this procedure has a

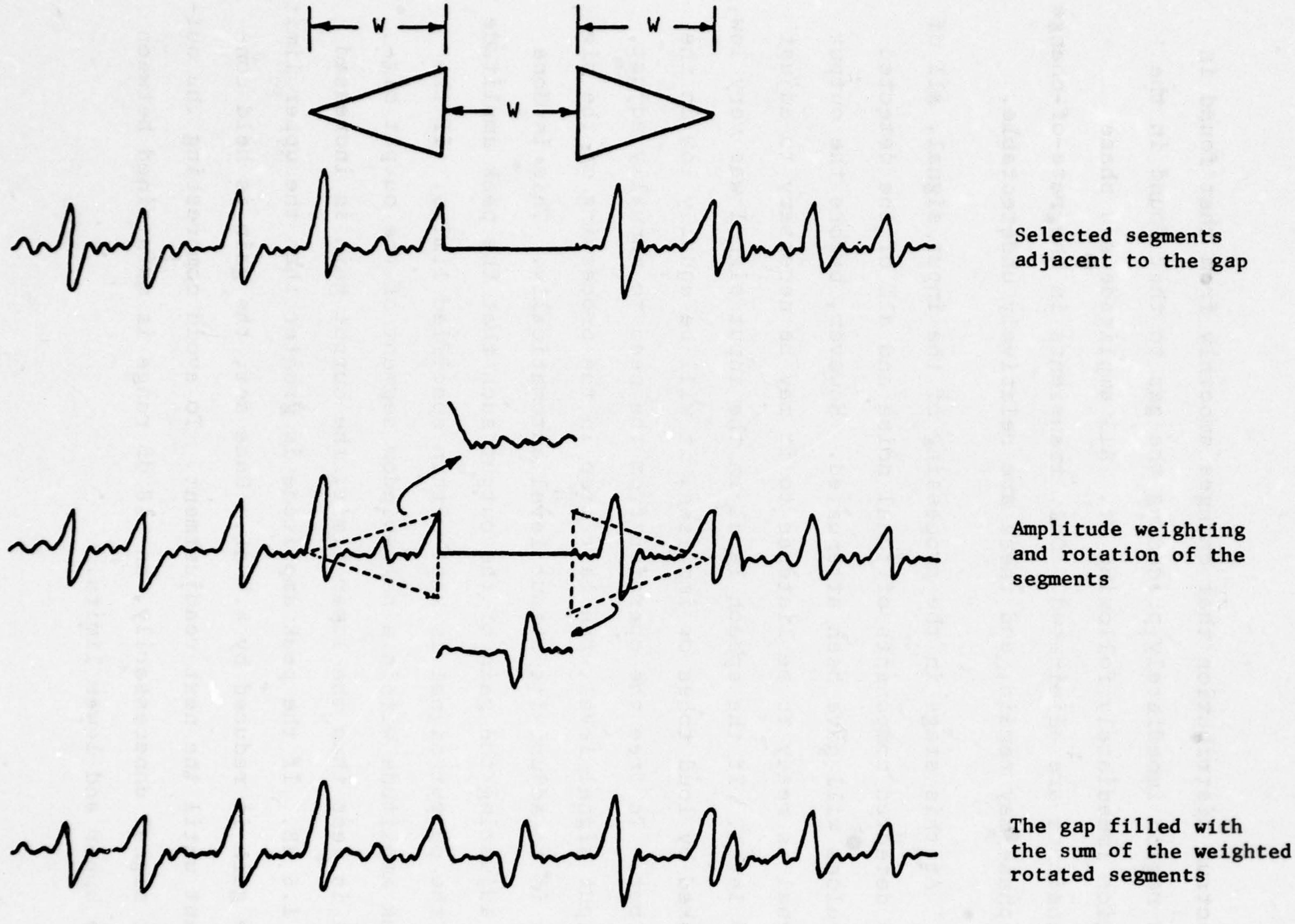


FIGURE 14 METHOD OF FILLING GAPS THAT RESULT FROM THE ELIMINATION OF IMPULSES

spectrum distribution that changes smoothly from that found in the region immediately preceding the gap to that found in the region immediately following it. All amplitude and phase transients are eliminated. Only transients in the rate-of-change of phase may remain, and these are relatively undetectable.

At this stage in the processing of the input signal, all of the detected components of tonal noise and all of the detected impulses will have been attenuated. However, before the output signal is ready to be listened to it may be necessary to adjust the level. If the speech level in the input signal was very low, masked by loud tones or impulses, it will be equally low in the output. To free the operator from the need to manually adjust output signal level, the last step in the processing of the signal is to adjust its output level automatically. This is done by adjusting the gain of the output such that the peak amplitude of the output signal is kept within specified limits. If the peak amplitude within a half-window segment of the output signal is less than the lower limit, the output gain is increased by 1.6 dB. If the peak amplitude is greater than the upper limit the gain is reduced by 4.4 dB. Once set, the gain is held constant until the next readjustment. To avoid compressing the output signal unnecessarily, an 18 dB range is maintained between the upper and lower limits.

4.0 HARDWARE COMPONENTS OF THE ADM

This section of the report describes the hardware components of the ADM and discusses the major considerations that led to their selection. The design ,specification, and selection of components was based on two somewhat independent groups of factors. In the first, and by far the most important group are those factors that are related to the need to meet the performance objectives of the ADM. Most prominent among these are real time operation, at least 40 dB attenuation of tones and impulses, and a speech signal dynamic range and bandwidth of at least 60 dB and 2500 Hz. In the second group are the factors that facilitated the development, implementation, and testing of the ADM. These include reliability, flexibility, ease of use, and for computer equipment, ease of programming. Although much less important than these, but also considered, were factors such as size and weight to the extent that they seriously affected the compactness and the mobility of the ADM.

In the discussions that follow, the function of each system in the ADM is described and related to the performance specifications the system must meet in order to satisfy the performance objectives of the ADM. Where appropriate, circuit designs are presented and discussed. For detailed descriptions of the circuits of major purchased components of the ADM the reader is referred to the manuals published by the manufacturers of those equipments.

4.1 The Input System

Before processing of the input signal can begin, the analog signal must be converted to digital form. The task of the input system is to provide the digital input data without either losing information contained in the signal or introducing distortion. These requirements led to performance specifications that were imposed on purchased components or circuit designs, as described below.

4.1.1 Anti-Aliasing Filtering

To avoid generating aliasing distortion components in sampled data, the sampling rate must be at least twice the highest frequency of significant components of the input signal. Typically, the bandwidth of speech communication channels is about 3000 Hz. However, significant components of speech or of noise are likely to be present at frequencies well above 3000 Hz. Consequently, the sampling rate must be higher than the 6000 samples per second needed to preserve the speech information in the 3000-Hz wide communication channel. It is desirable to keep the sampling rate as low as possible in order to minimize the amount of data to be processed. This is accomplished by using a low-pass filter to attenuate components above 3000 Hz as rapidly as is possible. For example, by using a filter with a cut-off attenuation slope of 140 dB per octave, the sampling rate need be only about 2.5 times the cut-off frequency, i.e., about 7500 samples per second in this case. However, such a filter will introduce audible ringing components in a speech signal. To minimize the risk of ringing, the sampling rate for the ADM was made 10,000 samples per second, which permitted the use of the filter whose characteristic is shown in figure 15.

4.1.2. Sampling and Analog-to-Digital Conversion

Since noise accompanying speech will raise the dynamic range of the combined signals over that of either one, to achieve a dynamic range of 60 dB at the output requires that the dynamic range be greater than 60 dB at the input. To establish a "worst case" design, we assumed that the input signal-to noise ratio could be as low as -18 dB. Again assuming a worst case situation, this requires the input dynamic range to be 78 dB. To satisfy this input requirement, the signal must be sampled and converted with 13-bit accuracy. A Datel Model ADC 149-14B unit, with 14 bit range and $\pm\frac{1}{2}$ LSB accuracy and linearity was selected to perform the analog-to-digital conversion. This device completes a 14-bit conversion in 50 microseconds, which is well within the required sampling interval of 100 microseconds.

An ideal sampler would take samples of the input signal instantaneously. In practice, even the best samplers have a finite, if very short, aperature time. Since the input signal is changing continuously, the amplitude of a sample is an

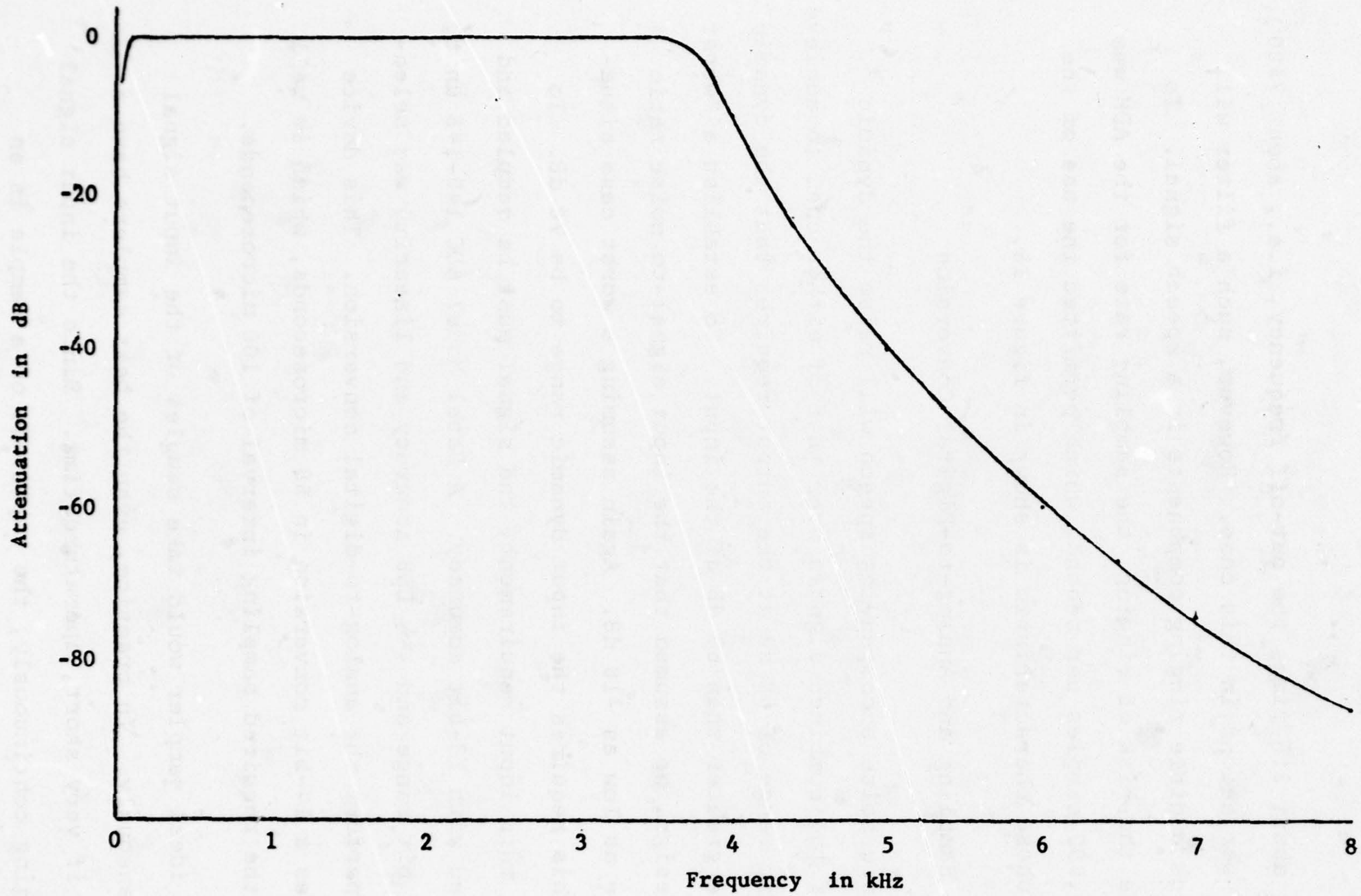


FIGURE 15 ATTENUATION CHARACTERISTICS OF THE ANTI-ALIASING FILTER

average of the signal observed over the aperature interval. To insure that the error in the sample is less than some minimum, the aperature must be less than some maximum value. The aperature required for the ADM sampler was determined by assuming that the input signal contained a 3000 Hz tone at a peak level of 4096 units (i.e., 12 bits). The maximum rate of change of this signal is $d_{max} = 2 \times 3000 \times 4096 = 77.2 \times 10^6$ units per second. At this rate, the signal will change by 1 unit (i.e., by 1 bit) in 13 nanoseconds. If a 13 nanosecond aperature is used, the sample value would be 0.5 units greater than the signal level at the start of the aperature interval and 0.5 units less than the signal level at the end of the interval. To keep the maximum sampling error less than ¼ bit, the maximum aperature must be less than 6.5 nanoseconds. If the objective is to limit the average sampling error to ¼ bit the maximum aperature is 10 nanoseconds. To achieve these levels of sampling accuracy, it is essential that jitter in the sampling aperature be negligible compared to the aperature time. Both requirements were met by an Optical Electronics Model 5021 Sample and Hold unit. The aperature time of this device is at most 3 nanoseconds, and the aperature jitter 0.3 nanoseconds. Moreover, the leakage current of this unit is less than 10 nanoamperes, which makes it very easy to hold the sample value constant to within ¼ bit during the 50-microsecond analog-to-digital conversion period. For example, by use of a 500 picofarad capacitor as the hold memory, the decay rate will not exceed 1 volt per second. Since the peak input voltage to the analog-to-digital converter is 5 volts, the maximum decay of 50 microvolts in 50 microseconds corresponds to a drift of one part in 10^5 or less than ¼ bit out of 13 bits.

Jitter in the sampling aperature also can be caused by jitter in the clock waveform that initiates sampling. As above, the requirement is for less than 0.65 nanoseconds jitter. The sampling rate being 10 kHz, this amounts to a clock jitter of 6.5 parts per 10^6, The required stability was achieved by use of a crystal oscillator to generate the clock waveforms.

4.1.3 The Input System Circuit

The input system is shown diagramatically in figure 16 and schematically in figure 17. Following the adjustment of level, input signals are amplified (IC1 and IC2) and filtered. The filtered signal is amplified again (IC3) and then applied to the sample-and-hold circuit (IC4). Every 100 microseconds this device samples (i.e., observes) the signal for a period of about 2 nanoseconds. The amplitude of the signal observed at each sample is "held" until the next sample is taken. These "held" voltages are transformed by the analog-to-digital (A/D) converter (IC5). As soon as each conversion is completed (about 50 microseconds after it was begun), the 14-bit output of the A/D converter is transferred to the MAP via registers (IC 6, 7, and 8).

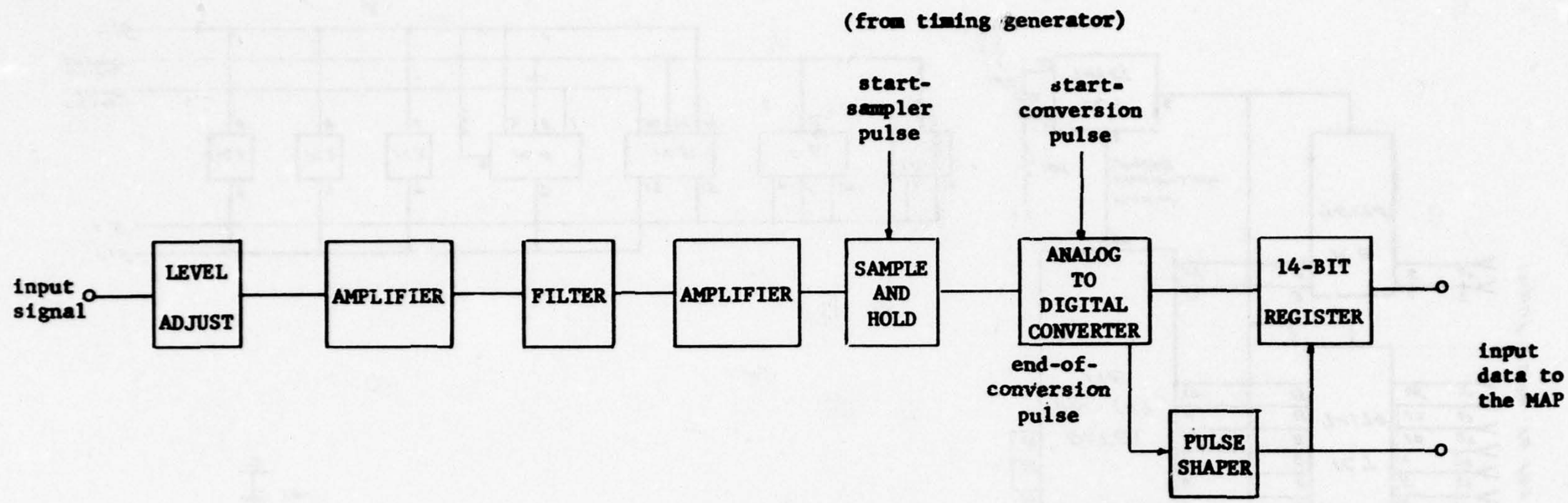


FIGURE 16 INPUT SYSTEM BLOCK DIAGRAM

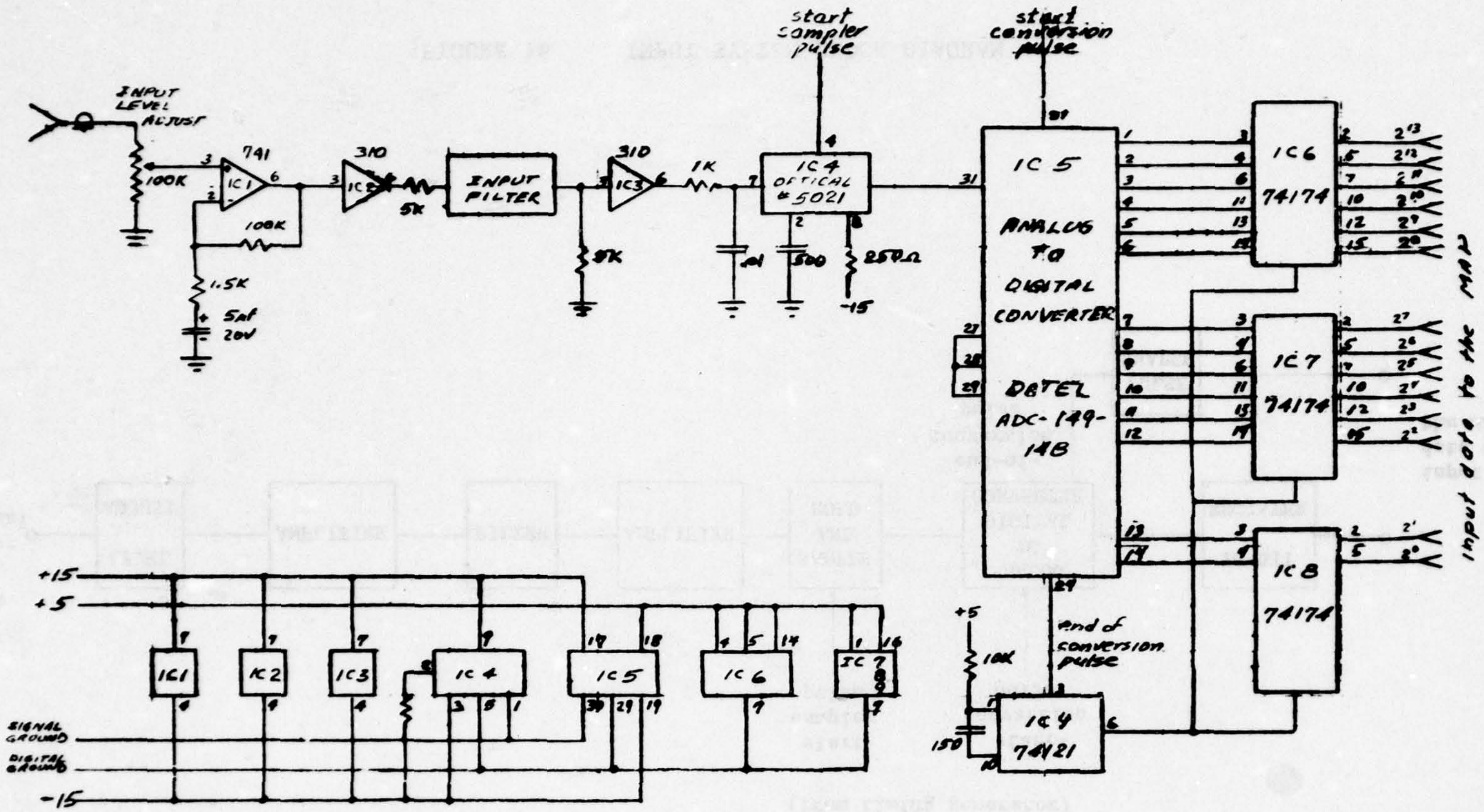


FIGURE 17 INPUT SYSTEM CIRCUIT DIAGRAM

Transfer of the binary data is initiated by an end-of-conversion pulse that is generated by the A/D converter and shaped by a one-shot multivibrator (IC9).

4.2 The Output System

The output signal is generated by the MAP in the form of a uniformly spaced sequence of samples represented by 14-bit binary numbers. These are provided to the output system at precisely the same rate as the sampling of the input signal, i.e., 10,000 samples per second. The output system converts the stream of digital data to a continuous analog waveform. The contract called for an output dynamic range of 60 dB, in binary terms, a range of 10 bits. To avoid any degradation of the signal by the output system, all 14 bits in each output sample are converted to analog form by a Datel DAC 169-14B digital to analog converter. This unit completes a 14-bit conversion in about 0.8 microseconds and, driven by a 14-bit "deglitching" latch, generates negligible switching transients.

The spectrum of the output waveform of a DAC is periodic, with peaks located at integer multiples of the sampling frequency and upper and lower sidebands centered at each peak. The sidebands at DC represent the spectrum of the regenerated output signal. The sidebands at harmonics of the sampling frequency are images of this spectrum. To insure that none of these appear in the output of the ADM, the DAC output waveform is passed through a low-pass filter whose characteristic is similar to that shown in figure 15.

The form of the output system is shown in figure 18 and the circuit in figure 19. To avoid distorting the regenerated speech signal, the spacing between samples of the output signal must be exactly the same as the spacing between samples of the input signal, i.e., 100 microseconds. This is accomplished by using the start-of conversion pulse to strobe the 14-bit binary data into registers (IC1, 2, and 3). The data are held by the registers between strobes, and are converted to DC current levels by the DAC (IC4). An amplifier converts these to voltage levels. The resulting discontinuous voltage wave form is filtered and then amplified (IC6). After adjustment in level, the signal is amplified by a power amplifier (IC7 and IC8) and delivered to the ADM output.

4.3 Computer for Processing the Signal

4.3.1 Speed Requirements

Two types of operations are performed in the signal: time and frequency transformations, and detection and attenuation of noise. These operations must be completed within half of the process period (i.e., within a half analysis window) if real-time operation is to be achieved. Otherwise, since the analysis

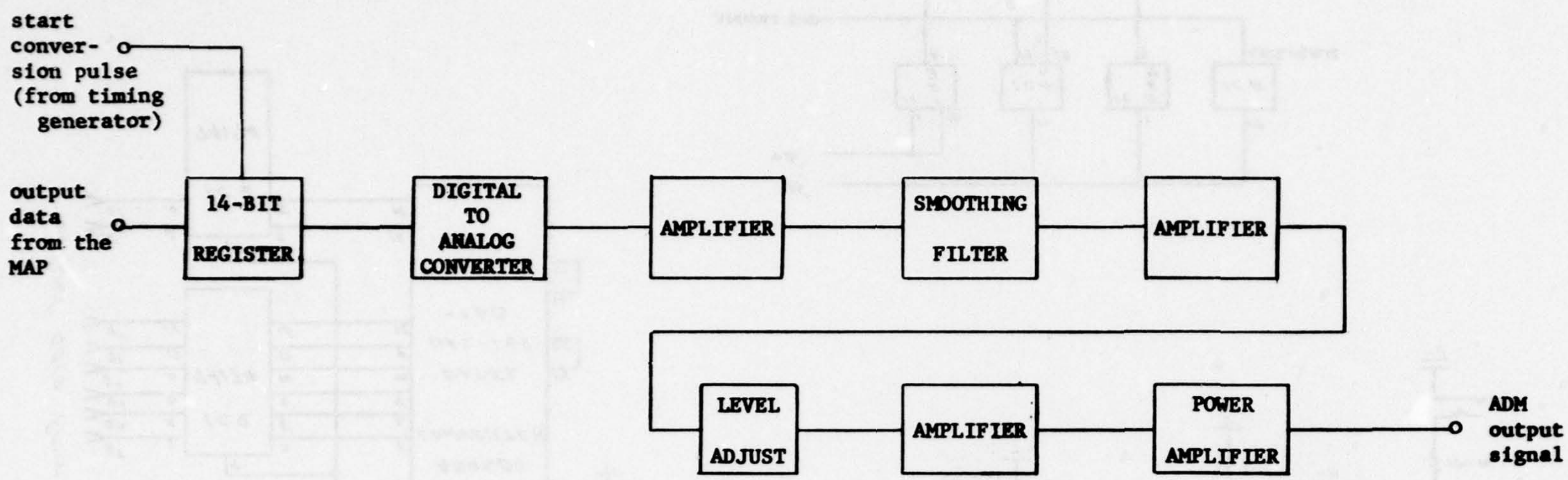


FIGURE 18 OUTPUT SYSTEM BLOCK DIAGRAM

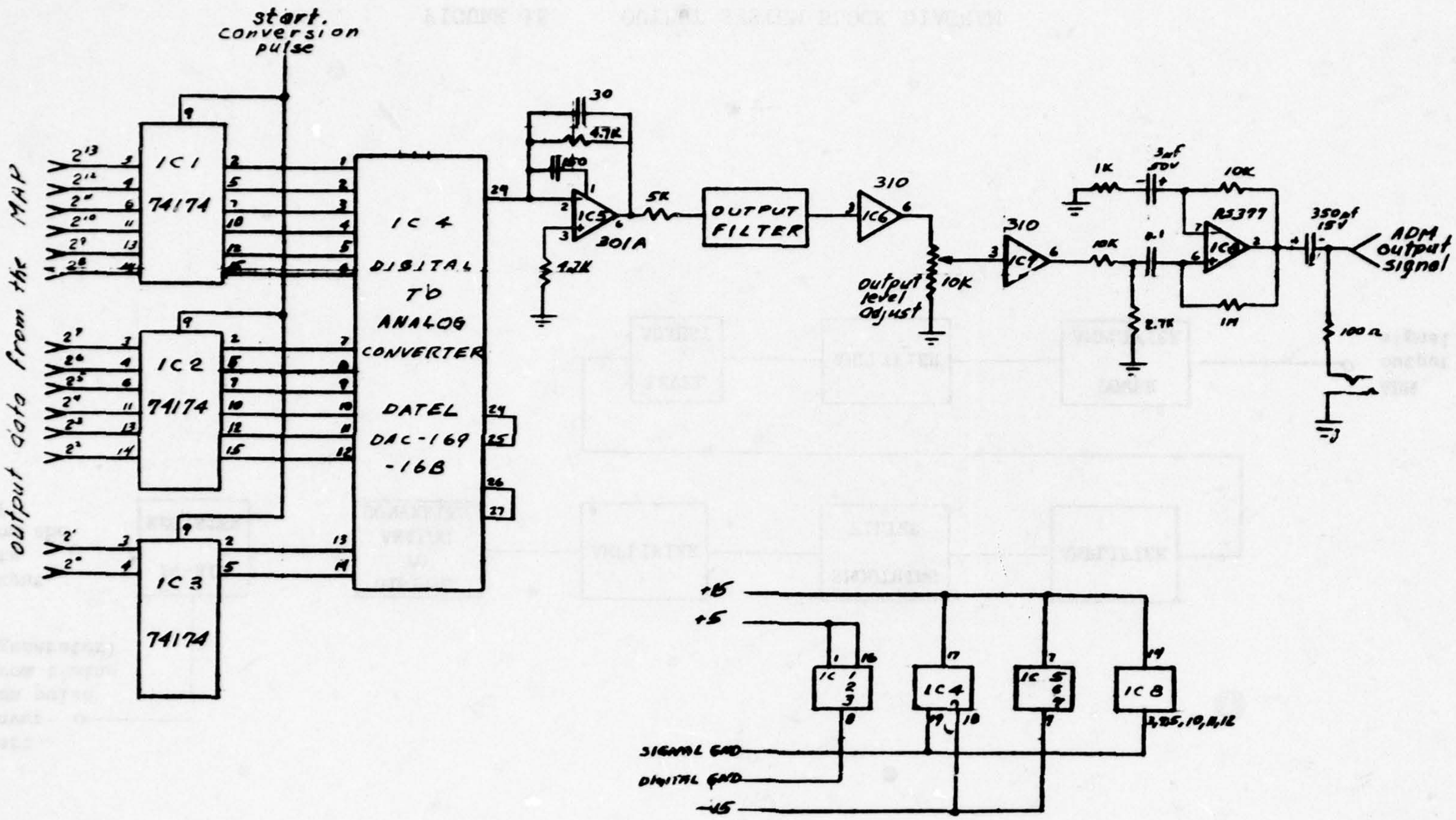


FIGURE 19 OUTPUT SYSTEM CIRCUIT DIAGRAM

window is moved in half-window length steps, a slip will occur between the input and output and an increasing amount of the input will fail to be processed. The speed required to achieve real-time operation was the major factor affecting the choice of a computer for use in the ADM.

To estimate the required computation speed, we assumed, based on computer simulation of the ADM, that the time and frequency transformations would take about half the available processing time, that is, one-half the process period (0.5P). Thus, within each one-quarter process period (0.25P) the computer would have to calculate two FFT's, each one in 0.125P. The number of points in each FFT is equal to the number of time samples in each analysis window, which is 10,000P. Therefore, the computer must be capable of calculating 10,000P points in 0.125 seconds, or 80,000 points per second.

The required computation speed rules out the use of general purpose minicomputers, (and for that matter, most large general purpose computers). Only a special purpose, high-speed device such as an array processor can satisfy the speed requirements. After examining the characteristics of several candidate machines we chose the MAP 200*, Manufactured by CSP, Inc., of Burlington, Mass. This machine is a very powerful minicomputer with an architecture and instruction repertoire that make possible very high speed processing of array data. A general description of

*MAP is the manufacturer's acronym for Macro-Arithmetic Processor.

this machine is given in Appendix A. Using FFT algorithms supplied by CSP, the MAP 200 can compute 97,500 spectrum or time points per second, which is more than 20 percent faster than that required. The MAP 200 can easily be upgraded to become the more powerful MAP 300, which is about three times faster. This feature makes it possible to implement additional processing techniques in the future, including the INTEL technique, without sacrificing real time operation or requiring gross changes in the components of the ADM.

In addition to calculating the transforms of the signal, the MAP also executes all of the operations in the tone and impulse noise detection and attenuation processes. The MAP also generates the signals that are needed for the display of the amplitude spectrums, the locations and widths of the attenuation gates, and an indication of the amplitude of the input signal. In between generating the spectrum and gate waveform displays, the MAP samples the switch settings in the control panel and the control panel setting for the location of the cursor gate. These operations could not have been performed by the minicomputer operating in parallel with the MAP or acting as the host computer since it would have required much more than the time available (a half analysis-window, as discussed earlier) to complete its tasks. But, the CSPU in the MAP (a processor similar to a CPU in most computers) was adequate for these tasks. To achieve real time operation of the ADM it was necessary to use the high speed arithmetic processor unit (the APU) in the MAP to perform almost all of the other program operations.

4.3.2 Computer Memory Requirements

The MAP 200 contains three separate, independent, asynchronously operating MOS memories that can be accessed through individual memory busses. Each memory is capable of storing 4096 32-bit words. The programs that constitute the software system of the ADM require about 2000 words of storage. The remaining 10,000 words of memory are used to store data. About 2000 words are required for the incoming and outgoing time waveforms, and another 2000 for the complex spectrum. An additional 1500 words are needed for the input signal overlap weighting and output signal overlap regeneration operations. The remaining 4500 words are required for storage of current-window and previous-window amplitude spectrums, complex-spectrum attenuation arrays, display waveform arrays, results of intermediate tests and analyses, and parameter tables. About five full 32-bit data words are generated for each sample of the input signal in an analysis window. Consequently, if, as has been suggested, the longest analysis window is made 400 milliseconds in a modified version of the ADM, it will be necessary to double the size of the memories in the MAP.

4.4 Computer Software Loading System

The MAP 200 normally is used as a peripheral device that supports the operations of a host computer. In the ADM, the MAP is used as the central computer and itself requires peripheral support equipment. Two kinds of peripheral systems are needed--I/O devices for transforming the time waveform to and from digital form (described in Sections 4.1 and 4.2), and devices for loading the MAP with the ADM programs.

Since the MAP uses volatile memories, all of the information stored in it is lost whenever the ADM is turned off. Consequently, the programs used in the MAP must be loaded into it each time the ADM is turned on. The system that is used to load the MAP consists of a digital tape recorder and a minicomputer. When operation of the ADM is initiated, the minicomputer reads the programs from digital magnetic tape and transfers them to the MAP. The tape recorder, a COI 3000D Linc Tape Unit, manufactured by Computer Operations, Inc., contains an auto-bootstrap that is activated by a simple contact closure. When a momentary contact switch on the control panel of the ADM is pressed, the auto-bootstrap causes the first file on the tape to be read by the minicomputer. This file contains the program that commands the reading of the rest of the tape and the transfer of ADM programs to the MAP.

The minicomputer is a Digital Equipment Corp. model PDP 11/04 with a 16K- word core memory. It contains the interface and controller circuits needed for communication with the Linc Tape Unit. Circuits for communicating with the MAP are contained in both the MAP and the minicomputer. An RS-232 interface circuit built into the I/O controller of the minicomputer permits it to transfer ASCII coded data to and from a teletypewriter or, if desired, a video terminal at rates up to 1200 baud. This feature was particularly valuable during the development and debugging of the ADM programs, when the minicomputer was used to read and/ or alter the contents of the MAP memories. The programs for accessing the MAP memories were stored on the Link Tape and were read into the minicomputer under command from a terminal.

4.5 The Display System

The display system provides the operator of the ADM with the information he needs to make best use of the manual procedure for detection of tones. This information consists of the amplitude spectrum of the input signal, the location and width of the cursor gate, and the locations and widths of the stored gates. The system also provides an indication of the level of the input signal and of whether the signal is within the range for optimum operation of the ADM. All of this data is presented in the form of a display on an oscilloscope screen.

The MAP generates X, Y, and Z axis display data for each item of information to be displayed. X and Y data are converted from 10-bit binary numbers to analog voltages and applied to the horizontal and vertical deflection inputs of a Tektronix model 604 Display Oscilloscope. Z axis data are provided as a 2-bit binary number and are converted to three analog levels for application to the intensity input of the oscilloscope. These levels correspond to CRT spot intensities of zero,(level of 0), dim (level of 1), and bright (level of 3).

A typical display is shown in figure 20. The MAP generates a complete set of display waveforms eighty times each second, producing a fully flicker free display. As shown in this figure, the amplitude spectrum is displayed as the top-most pattern on the oscilloscope screen. The frequency range is from DC to 3000 Hz. Below it can be seen a trace that shows the location and width of the cursor gate, which is displayed as a positive or upward going step. In this figure, the gate is seen to be at a position corresponding to a frequency of about 1000 Hz on the amplitude spectrum, and is about 190 Hz wide. The spectrum region immediately above the cursor gate is dimmed by the MAP to indicate the presence of an active gate over that frequency range. Previous positions of the cursor gate that were entered into the ADM memory are shown on the same trace as downward going or negative steps. Here too, the spectrum trace is dimmed at locations that correspond to those of the gates.

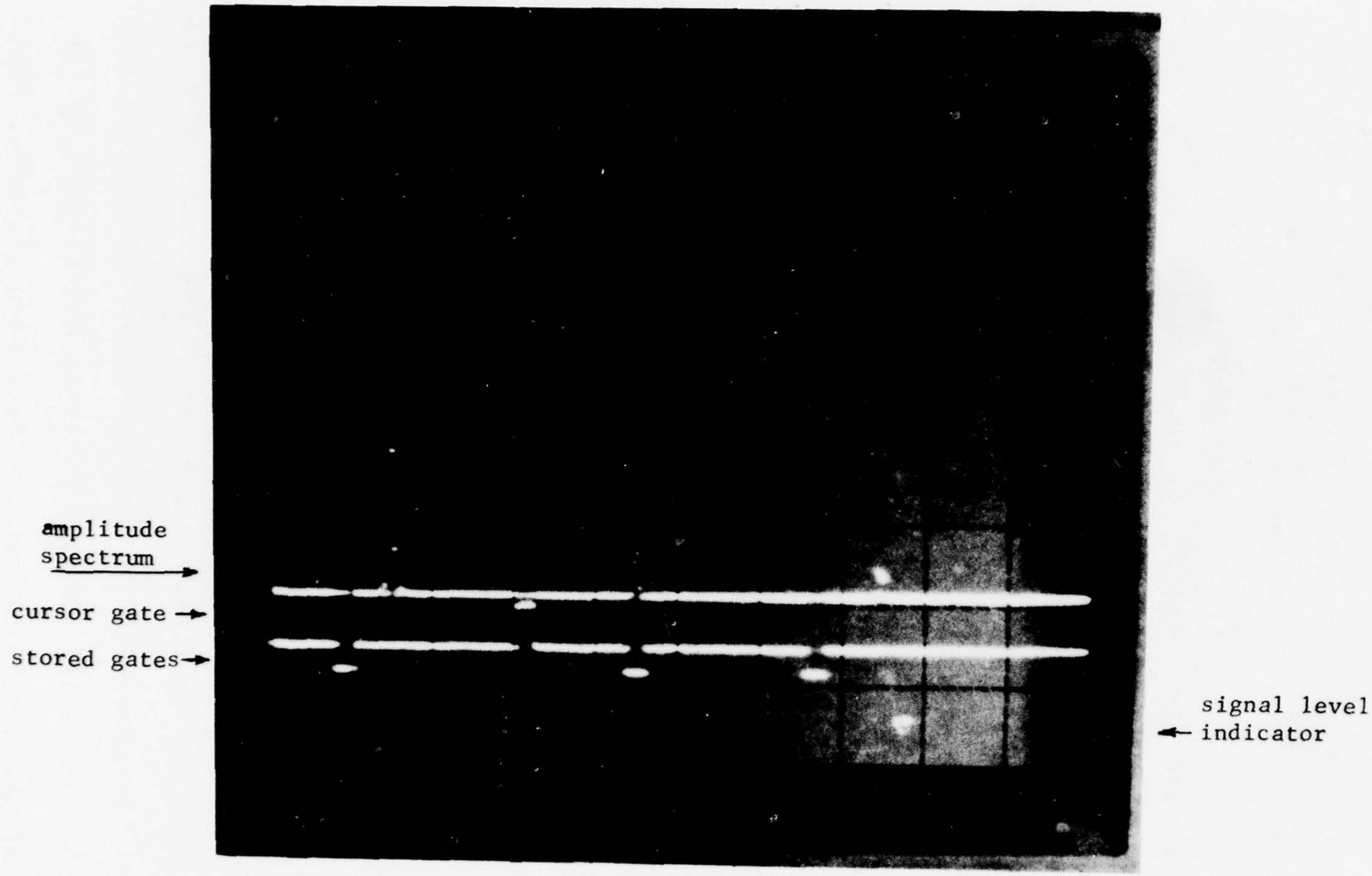


FIGURE 20 TYPICAL DISPLAY OF ADM DATA

At the bottom of the display is a downward pointing arrowhead. This is the signal level indicator or SLI. The displacement of the SLI from the left-hand edge of the display is proportional to the average level of the input signal. As the signal level varies, the position of the SLI will vary in a like manner, with a time constant of about 350 ms. For signals that are too weak for optimum performance of the ADM (below 280 units as sampled), the SLI will appear at the left hand edge and will be dimmed. For signals that are too strong (greater than 1600 units) the SLI will be at the right hand edge and dimmed.

The minimum and maximum limits on the signal level were determined as follows. The minimum limit corresponds to the average level of an input signal that contains a tone and, 18 dB below it, a speech signal whose peak level is 36 dB below the maximum of the analog-to-digital converter in the input system. In terms of bits, the peaks of the speech signal are at a level of 7 bits on a scale of 13 bits. For this speech signal, the average level would be at a level of 5 bits. The average level of the accompanying tone would be about 3 bits above that of the speech, or about 8 bits on a scale of 13 bits. Equivalently, the average level of the tone would be about 256 units on a scale of 8192 units. To provide a margin of safety, the minimum limit was set at 280 units. The maximum limit is established by the requirements of the automatic impulse detection process. Since the impulse detection threshold is computed as five times the average signal level, the largest average signal level for which impulses can be detected is 8192/5 units, or 1638 units. To provide a margin, the SLI is dimmed when the average signal level is above 1600 units.

The circuit of the display generator is shown in figure 21. The MAP transmits X, Y, and Z axis data to the display system at a rate of 102,400 displayed points per second. First, a 10-bit number for the horizontal position of the CRT spot is received from the MAP and strobed into a register (IC1 and IC2). About 9 microseconds later the vertical position and intensity data are received. These are strobed into latches IC5 and IC6 and, at the same time, the horizontal position data are transferred to latches IC3 and IC4. Digital-to-analog converters IC7 and IC8 (Datel model DAC-HZ-12BGC) transform the horizontal and vertical data to DC currents and these are transformed to voltages by IC9 and IC10. The 2-bit intensity data are converted to levels by a summing operational amplifier, IC11. Less than one microsecond later, the next X-axis value is delivered to the display generator. Since the X, Y, and Z latches are strobed only with the arrival of the Y and Z data, the outputs of the DACs are held for the full 9.76 microsecond interval between strobes.

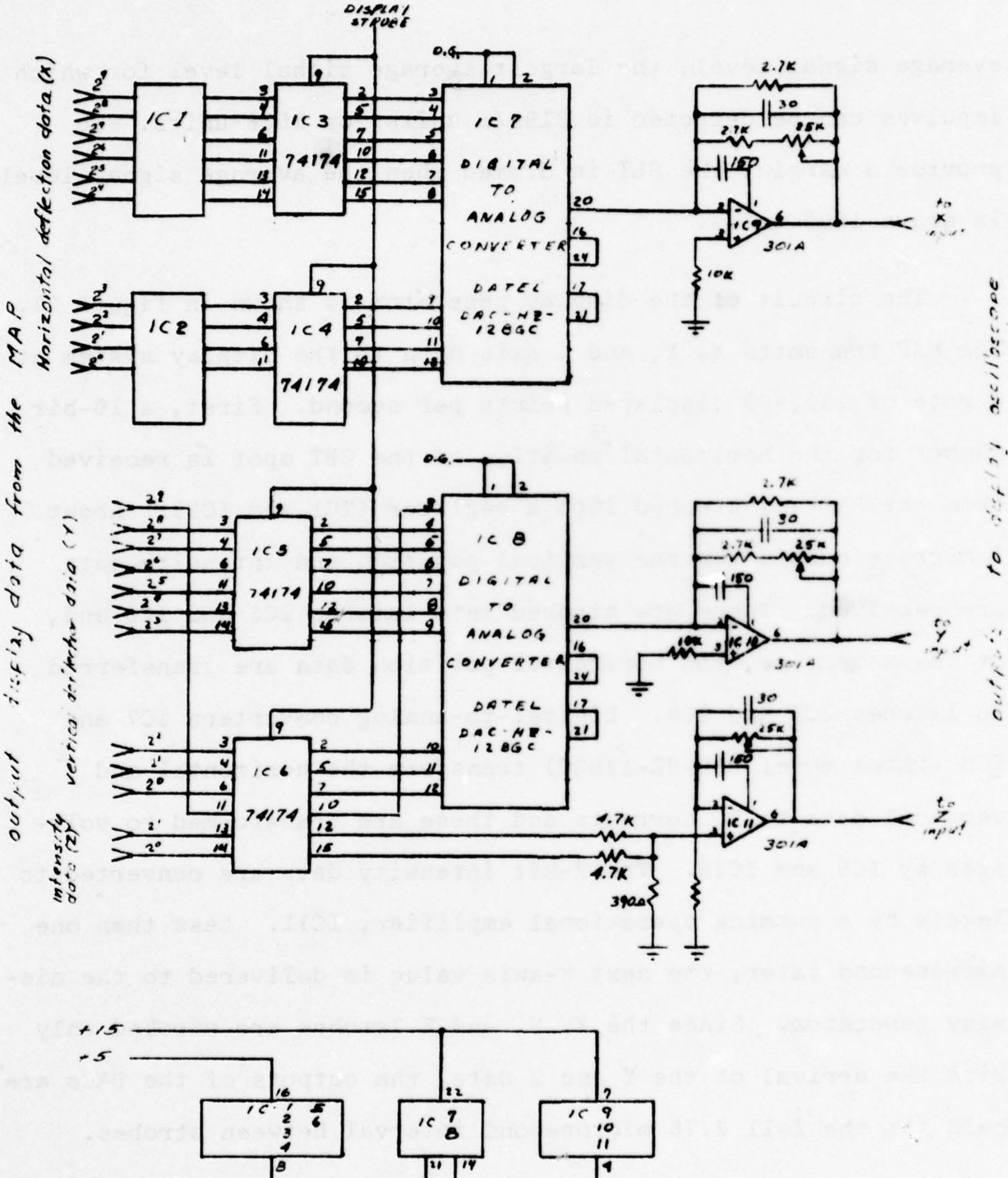


FIGURE 21 DISPLAY GENERATOR CIRCUIT DIAGRAM

4.6 The ADM Control Panel

The user of the ADM must make certain necessary adjustments to obtain optimum performance from the system. Among these are the setting of the input signal level, the selection of the process period, and the enabling or disabling of the automatic processes. In addition, the user must enable and control the operation of the manual tone detection process. All of the controls needed for these operations are located on a panel on the front of the ADM.

4.6.1 Functional Description of Controls

The control panel is shown in Figure 22. For the convenience of the user, the controls are grouped according to function into three groups: signal control, automatic process control and manual process control. The function and effective use of each of the controls is described as follows.

1. Signal control group.

   INPUT LEVEL. This control is used to adjust the level of the incoming signal. It should be set so that the signal level indicator (the SLI that is displayed on the screen of the oscillope screen) swings into the upper half of its range. With the level control set to maximum, a 0.1 volt input signal will drive the ADM to its full dynamic range.

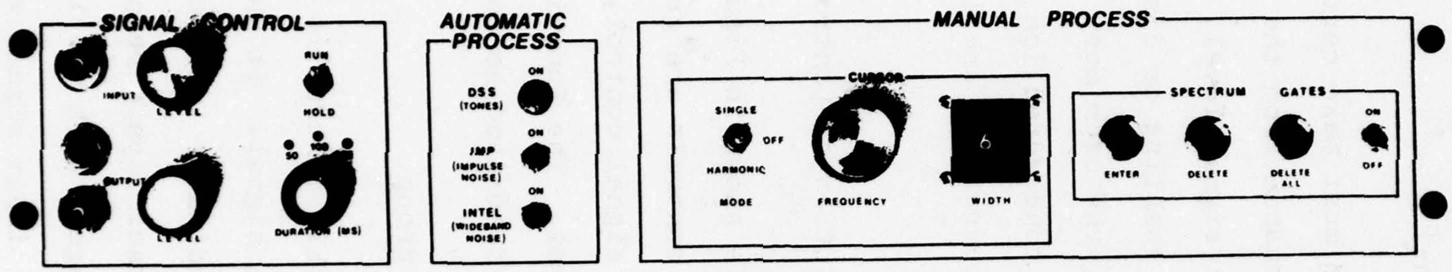


FIGURE 22 LAYOUT OF THE CONTROL PANEL

OUTPUT LEVEL. This control adjusts the level of the output signal. The maximum output level, 3 volts rms into a load of 8 ohms, is adequate to drive headphones or a small loudspeaker.

DURATION. The duration of the process period is set by this 3-position rotary switch. The duration of the process period should be set initially at 100 msec. If the input signal contains noise that is stable in amplitude and frequency the duration should be raised to 200 msec. On the other hand, if the noise consists of components that change rapidly or are of very brief duration the duration of the process period should be made 50 msec. Table 4 provides a guide to the optimum setting for various rates-of-change of frequency and for various durations of noise components. However, the most reliable guide is the ear. When speech is not detectable in the output of the ADM the best setting is the one that produces the greatest attenuation of the noise. This setting maximizes the ability to detect the presence of speech that is too weak to be heard. Once speech is detected, the duration of the process period should be adjusted until the speech is easiest to understand. These two criteria-detectability and intelligibility of speech - usually lead to the same setting for the duration of the process period.

Under some circumstances (e.g., for some combinations of stationary and variable noises) the settings will be different. Therefore it is best to follow the two-step procedure outlined above when speech is not detectable in the output of the ADM.

Table 4

OPTIMUM DURATION OF THE PROCESS PERIOD

Duration of Noise Component (msec)	Rate-of-Change of Component Frequency (Hz/sec)	Optimum Duration of Process Period (msec)
>200	<100	200
100 - 200	100 - 400	100
<100	>400	50

<u>RUN/HOLD</u>. This 2-position toggle switch permits the operator to "freeze" a segment of the input signal. In the RUN position the input signal is admitted continuously to the ADM. Successive process-period length segments are processed in the order in which they arrive. In the HOLD position the segment currently being processed is not replaced by the succeeding one but is stored and reprocessed repeatedly. By holding a segment, the operator can hear it again and again and can examine its spectrum carefully. This feature is particularly valuable when it is necessary to manually detect tones that are too weak or too intermittent to be

detected any other way.

2. Automatic Process Group.

This group consists of three 2-position toggle switches. Two of them are used to turn on or off the automatic DSS and IMP processes. The third switch will control the INTEL process for attenuating wideband random noise when that process is implemented in the ADM.

3. Manual Process Group.

CURSOR MODE. In the middle position of this 3-position toggle switch, the cursor gate is turned off. It will disappear from the display and will not attenuate components of the input signal. In the upper position (SINGLE), the cursor gate is active and appears on the display. Components of the signal whose frequencies fall within the cursor gate will be attenuated. In the lower switch position (HARMONIC) the cursor gate is repeated at uniformly spaced frequency intervals. This setting of the switch permits the operator to place cursor gates at the frequencies of harmonics for a periodic noise, such as a buzz or power line hum.

CURSOR FREQUENCY. This control is used to adjust the location the cursor gate. In practice, the operator should adjust the cursor gate loaction while observing the displayed amplitude spectrum and listening to the output signal. The cursor gate location should be adjusted carefully for maximum attenuation of the unwanted tone.

GATE WIDTH. The width of the cursor gate can be adjusted in ten uniformly spaced steps by use of this 10-position lever switch. The range of gate widths for each selectable process period duration is given in Table 5.

Table 5

FREQUENCY INCREMENTS AND RANGE
OF SETTINGS OF THE CURSOR GATE WIDTH

Duration of Process Period (msec)	Change in Width at Each Step (Hz)	Range of Cursor Gate Widths (Hz)
200	10	15 - 105
100	20	30 - 210
50	40	60 - 420

Once the location of the cursor gate has been set as described above, the width of the gate should be adjusted until the desired degree of attenuation is achieved. To minimize the risk of attenuating components of speech, the cursor gate should not be made wider than necessary.

ENTER. The operator can store the location and width of the cursor gate in the ADM memory by pressing the ENTER button. The ADM will acknowledge that the gate data have been stored by reversing the direction of the cursor gate on the display. It will now appear as a downward-going step. Once a cursor has been stored, the cursor mode switch can be set to OFF or the cursor gate can be moved to another location. In either case, the stored gate will continue to attenuate signal components at the spectrum location at which it was entered.

Occasionally, the operator will need to attenuate a range of frequencies that is wider than the maximum setting provided by the gate-width control. He can accomplish this by entering a series of overlapping gates that cover the desired frequency range.

DELETE. Once a stored gate is no longer needed (e.g.,when the component it was attenuating disappears from the input) it can be removed from the ADM memory. This is done by placing the cursor gate at the location of the gate to be eliminated, adjusting the width of the cursor gate until it completely covers the stored gate, and then pressing the DELETE button. The same procedure can be used to narrow an extra-wide gate by deleting only the unwanted region of the stored gate.

DELETE ALL. On occasion, the operator may want to remove all the gates that were stored in memory. This can be done quickly and easily by pressing the DELETE and DELETE ALL buttons simultaneously.

GATES ON/OFF. This 2-position toggle switch must be in the ON position when it is desired to store or delete cursor gates. Gates that have been stored will be active when the switch is ON. By putting the switch to OFF, the gates will be disabled. By disabling and then reactivating the stored gates the operator can determine quickly if they are still needed.

4.6.2 The Control Panel Circuit

The wiring of the control panel switches and of the cursor-location control is shown in figure 22.

The control settings are read by the MAP as binary numbers. A toggle switch produces a binary 1 in the ON position, a binary 0 in the OFF position. Because a push-button switch makes only momentary contact when it is pressed, it is necessary to store such an event in a 1-bit memory. This is achieved by having each push-button switch "set" a flip-flop (IC3 and IC4) to a 1 any time it is pressed. The flip-flops are reset to 0 shortly after they are read by the MAP. The settings of seven of these switches, four of the toggle switches, and the three push-button

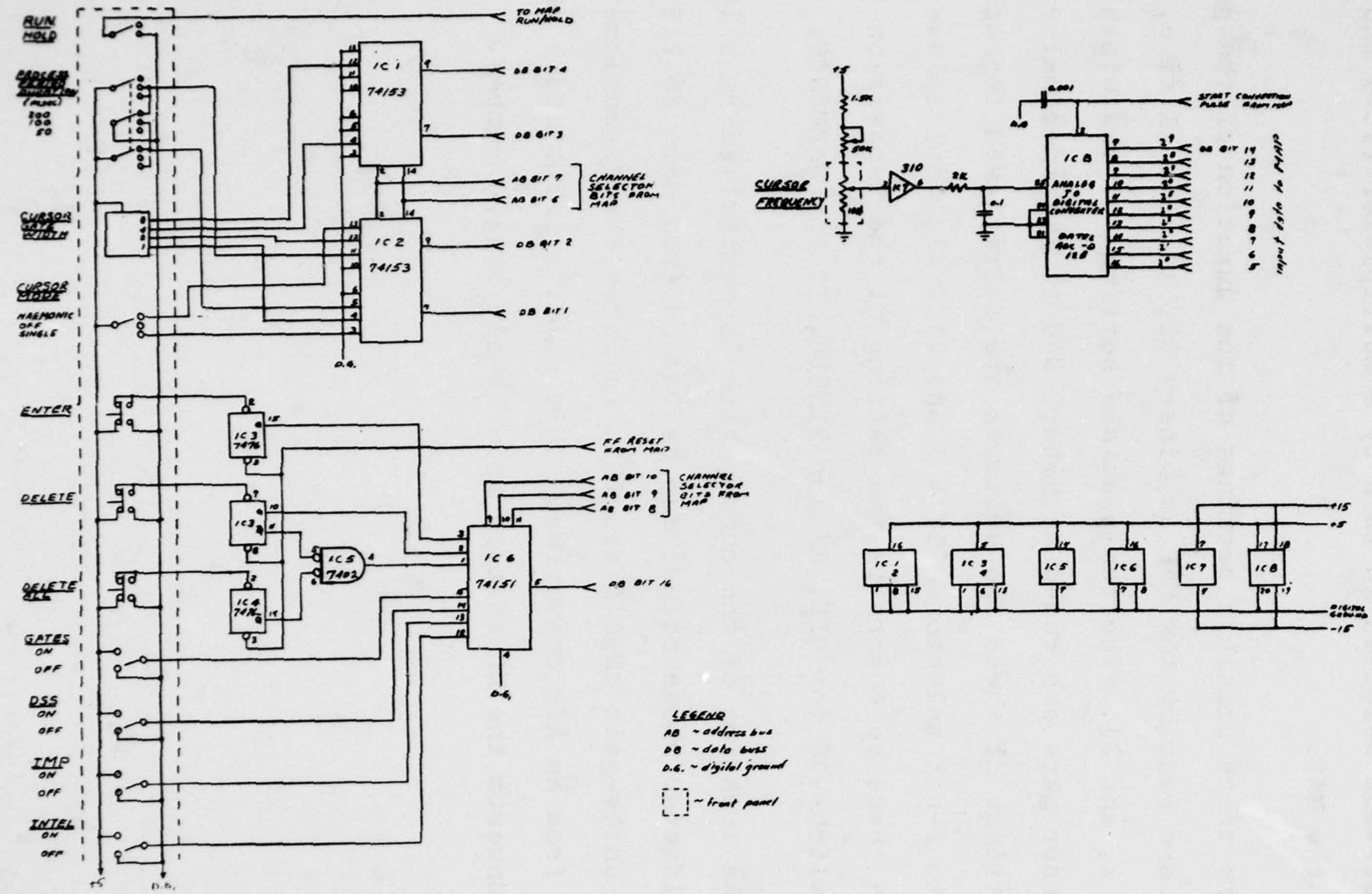


FIGURE 23 CONTROL PANEL CIRCUIT DIAGRAM

switches--are selected in turn by a multiplexer (IC6) and fed out to the MAP.

The three possible settings of the duration of the process period are read by the MAP as binary 01, 10, and 11 (i.e., decimal 1, 2, and 3). The ten possible settings of the width of the cursor gate are coded as binary 000 to 1001 (decimal 0 to 9). The settings of these two switches are multiplexed through two 4-bit to 2-bit selectors (IC's 1 and 2). Also multiplexed through these selectors is the setting of the 3-position cursor mode switch--00 for OFF, 01 for SINGLE, 10 for HARMONIC.

The location of the cursor gate is controlled by a 100K ohm potentiometer whose output can be varied from zero to 2.5 volts DC. A unity-gain amplifier, IC7, isolates the potentiometer output from an A/D converter. This unit, a DATEL model ADC-D 12B, converts the DC voltage to a 10-bit binary number.

5.0 ADM SYSTEM SOFTWARE

The software system of the ADM consists of two groups of programs. The main group comprises the routines that process the input signal as described in Sections 2, 3, and 4. The second group consists of the program that control the input and output of signal, display waveform, and control panel data. In this section of the report we describe the significant features of the data control programs. We then describe the overall system of signal processing programs, with primary emphasis on the sequence in which the operations on the signal are performed. Finally, the three programs that process the signal to detect and attenuate noise are described in greater detail.

5.1 Peripheral Control Programs

The input and output of signal, display, and control panel data are controlled by two I/O scrolls that are connected to the memory on bus 2. These I/O controllers operate independently and asynchronously of the memory bus and of other programmed operations being performed on the bus. The occasional simultaneous contention for the bus by two of the devices connected to it (these are the two scrolls, the APU, and the CSPU) are resolved by a scheduler that is part of the bus architecture. The two scrolls are referred to as the time-function scroll and the display/control scroll. The programs that control their operations are stored in local memory on the scrolls.

These programs set up the direction of the flow of data--into or out of the MAP--and the address in bus 2 memory to or from which the data will flow.

5.1.1 Input and Output of Signal Data

Incoming samples of the signal are loaded into one of two areas of bus 2 memory, which are referred to as input buffers. The samples are switched from one buffer to the other as soon as the buffer that is being loaded is full. Each buffer stores one-half the number of samples in an analysis window. The analysis window itself is formed of two halves. The upper half (which contains the most recent signal data) contains the samples that are stored in the currently full input buffer. The lower half contains the samples that were stored in the buffer that previously was full and now is being loaded with new samples. The data in an analysis window are assembled as follows. As soon as an input buffer is full, the incoming samples are shifted to the alternate input buffer. The contents of the upper half of the analysis window (i.e., the previous full buffer) are then shifted to the lower half and are replaced by the contents of the newly filled buffer (i.e., the current full buffer). In this way, the analysis window is moved in half-window length steps through the incoming time function data.

The output data also are loaded into two buffers. While one is being filled the data in the alternate buffer are read out by the time-function I/O scroll. To make it possible to use the same scroll for input and output of time function data,the input and output of samples of the signal are interlaced in time. Thus, each input sample is followed 50 microseconds later by an output sample. The scroll program is used in conjunction with scroll circuitry to direct the flow of the data sample, either taking it from the input analog-to-digital converter or else directing it to the output digital-to-analog converter.

When the buffer to be used for input or output of data is to be switched, the starting address of the new buffer and the number of samples it contains are loaded into registers in the scroll. Thereafter, the address is incremented and the number of samples decremented on the transfer of each successive sample. When the number of samples has been decremented to zero the scroll notifies the MAP that the loading of a buffer has been completed and requests a new buffer address and count of samples. Since the input and output of samples are interlaced in time, it was convenient to also interlace them in the buffers. Therefore, the number of memory locations required by a buffer is twice the number of samples contained in a half analysis window, with input and output samples alternating in the buffer. This scheme makes it unnecessary to provide two buffer starting-addresses to the scroll, one for an input buffer and the other for an output buffer. By incrementing the one starting address upon the completion of each transfer of a sample, the correct address of the next sample, whether input or output, is automatically generated. All that is necessary is for the scroll program to switch the flow of data such that odd numbered samples are input samples and even numbered samples are output samples.

5.1.2 Input of Control Panel Data and Output of Display Data

The display/controller scroll program repeatedly displays the amplitude spectrum, the manual gate array, and the signal level indicator (SLI). Between the display of the SLI and the amplitude spectrum the program samples the settings of the switches on the control panel and stores these as-1's or 0's in assigned locations in bus 2 memory. It also reads the 10-bit digitized setting of the cursor loaction control and stores this as the location of the cursor gate, which is expressed directly as the number of the spectrum sample, on a range 0 to 600, at which the cursor gate is to be centered.

For the display of the amplitude spectrum the scroll receives the starting address of the YCRT array and the number 600, which is the number of points in the display. To generate the horizontal coordinate data for the display, the scroll program increments a counter upon the display of each successive point. The display/control scroll receives,from the display generator requests for display data at a rate of 102.4 thousand times per second. Two successive requests are required for each displayed point. Upon receiving the first request the scroll increments and transmits the horizontal coordinate data. At the second request it reads the value of YCRT on bus 2 at the address currently stored in the scroll, and transmits this value to the display generator. The scroll then increments the YCRT address and decrements the count of transmitted data points.

The transmission of display data continues as described above until the count of transmitted data points reaches zero. Then the scroll is loaded with the starting address of the GCRT array, the number of transmitted data points is reset to 600,and the horizontal coordinate counter is reset to zero. The transmitted data points now produce the display of the manual gate array.

When the display of the manual gate array is completed, the scroll receives the starting address for the SLI data and the number 6, which is the number of displayed points in the SLI pattern. For this function, both the horizontal and the vertical coordinate data are obtained from the memory on bus 2. Following the display of the SLI, the control panel settings are read into the MAP. Then the display/control scroll is reinitiated for display of the amplitude spectrum and the cycle of waveform displays is begun again.

For convenience in reading the settings of the controls on the control panel into the MAP, each control is considered to be a device that is connected to the display/control scroll and is assigned a device number. The setting of each such device is read in rapid succession, starting with the digital number associated with the location of the cursor gate and ending with the impulse-process-enable switch. A total of 12 device settings are read and stored in successive locations in bus 2 memory starting at an initial address that is supplied by the MAP. The control panel settings are read once every complete display cycle, i.e., at a rate of about 80 times per second. Consequently, the response to a change in a setting is virtually "instantaneous."

5.2 Overall Description of the ADM Signal Processing Software System

The major steps and programs in the ADM signal processing software system are shown in figure 24. In general, the sequence follows the sequence of operations for processing input signals, as described in Sections 2 and 3. For some of the steps the sequence is designed to minimize the time required for execution of all of the operations. Wherever possible, operations were performed in parallel by the arithmetic processor unit (APU) and by the central signal processor unit (CSPU).

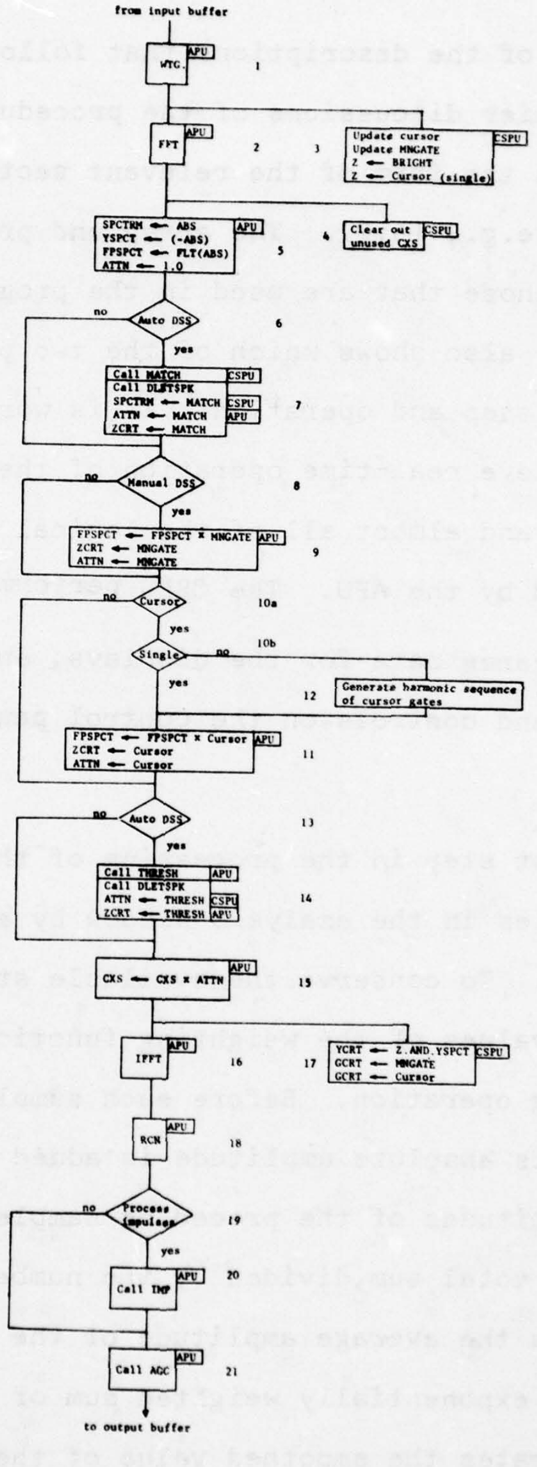


FIGURE 24 FLOWCHART OF THE ADM SOFTWARE SYSTEM

In some of the descriptions that follow, references are given to earlier discussions of the procedures or programs being described, in the form of the relevant section number enclosed in brackets, e.g., (5.2). The array and program names shown are the same as those that are used in the programs being described. The flowchart also shows which of the two processors is used to perform each step and operation. It is worth noting that in order to achieve real-time operation of the ADM, all of the calculations and almost all of the logical operations on the data are performed by the APU. The CSPU performs the peak matching routine, prepares data for the displays, and tests the settings of switches and controls on the control panel.

Step 1.

The first step in the processing of the signal is to multiply the samples in the analysis window by a symmetrical triangular function. To conserve the available storage capacity of the memory, the values of the weighting function are computed during the weighting operation. Before each sample of the signal is multiplied, its absolute amplitude is added to the sum of the absolute amplitudes of the preceding samples in the analysis window. The total sum, divided by the number of samples in the window, yields the average amplitude of the input signal. This, added to the exponentially weighted sum of previous average levels, generates the smoothed value of the average level of the signal that is displayed as the position of the SLI (4.5).

Step 2.

The second step is the computation of the complex spectrum of the weighted input signal by use of an FFT algorithm. The FFT program, originally written by CSP, Inc. for use with the MAP, was modified by us to remove all unnecessary,general purpose operations.

Step 3.

While the FFT is being computed by the APU, the CSPU performs several "bookeeping" operations. First it updates the values for the location and width of the cursor gate. If it finds that the ENTER button on the control panel was pressed since the last observation of it, the cursor data are entered into the array of stored locations of manually entered gates (MNGATE). All values in the array of values for control of the intensity of the spectrum display (ZCRT) are set to 3, which is the "bright" level (4.5). Finally, the ZCRT data in the region of the cursor gate are set to 1, (the dim level).

Step 4.

In this step the samples of the complex spectrum at frequencies above 3000 Hz are set to zero. Since the upper limit of the speech band of interest is 3000 Hz, the tone detection processes operate only up to that frequency. This step eliminates any tones that may be present at higher frequencies.

Step 5.

The absolute amplitude spectrum is computed from the complex spectrum in both fixed point (SPCTRM) and floating point (FPSPCT) forms. These are used by the automatic tone detection processes. A version of the SPCTRM array with all signs made negative is stored for the display of the amplitude spectrum in the YSPCT array. Finally, the values of the array ATTN are initialized to 1.0. This array is used later to multiply the complex spectrum to attenuate the spectrum components of tones.

Step 6.

The setting of the control panel switch that is used to enable the automatic tone detection processes is examined. If the stored value is negative, the next step is skipped.

Step 7. (Detailed in Section 5.3.1)

The locations and amplitudes of peaks on successive spectrums are compared to identify components of tones (2.2.1). Attenuation gates are generated at the locations of detected tones by the routine labeled DLET$PK (2.3). At corresponding locations, the amplitude spectrum array SPCTRM is set to zero, the spectrum attenuation array ATTN is set to 0.01, and the spectrum display intensity array ZCRT is set to 1 (dim).

Step 8.

The setting of the control panel switch that is used to enable the manual tone detection process is examined. If the stored value is negative, the next step is skipped.

Step 9.

The locations of manually entered attenuation gates, stored in the MNGATE array, are determined. At corresponding locations, the floating point amplitude spectrum FPSPCT is set to 0.0, the attenuation array ATTN is set to 0.01, and the spectrum display intensity array ZCRT is set to 1.

Step 10.

The setting of the three-position control panel switch that selects the mode of the cursor gate is examined. If it is negative the next two steps are skipped. If it is zero or greater the switch setting is tested to determine whether a single gate or a harmonic series of gates is desired. If the stored value is zero or greater the next step is skipped.

Step 11.

At the location of the cursor gate the FPSPCT array is set to 0.0, the ATTN array to 0.01, and the ZCRT array to 1. The program then continues with step 13.

Step 12.

A sequence of attenuation gates is generated with widths equal to that of the cursor gate and at locations that are integral multiples of the location of the cursor gate. These gates are applied to the FPSPCT, ATTN, and ZCRT arrays as in Step 11.

Step 13.

As in Step 6, the setting of the switch that enables the automatic tone detection processes is examined. If it is negative, the next step is skipped.

Step 14. (Detailed in Section 5.3.2.)

In this step, amplitude thresholds are calculated and used to automatically detect tones in the amplitude spectrum (2.2.2). At the location of each detected tone, the DLET$PK routine generates an attenuation gate of the appropriate width. These gates are applied to the ATTN and ZCRT arrays as in Step 7.

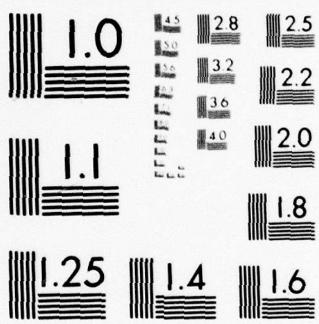
Step 15.

The complex spectrum, stored in the CXS array, is multiplied point-by-point by the ATTN array. The terms in the ATTN array are either 1.0, where the components of the spectrum are to be left as is, or 0.01, where they are to be attenuated, as determined in Steps 7, 8,11, 12, and 14.

AD-A062 844 QUEENS COLL FLUSHING NY DEPT OF COMPUTER SCIENCES F/G 17/2
THE SPEECH ENHANCEMENT ADVANCED DEVELOPMENT MODEL.(U)
NOV 78 M R WEISS, E ASCHKENASY F30602-76-C-0033
UNCLASSIFIED RADC-TR-78-232 NL

2 OF 2
AD
A062 844

END
DATE
FILMED
2-79
DDC



MICROCOPY RESOLUTION TEST CHART

Step 16.

The processed complex spectrum is transformed to a time domain signal by use of an inverse FFT algorithm.

Step 17.

While the IFFT is being computed by the APU, the CSPU prepares the spectrum and gate data for display. The display intensity data in ZCRT is merged with the amplitude spectrum data in YCRT to form a single array. The display hardware separates these display functions, as described in Section 4.5 (see figure 21). The stored manual gates, in the MNGATE array, and the cursor gate are loaded into the gate display array, GCRT.

Step 18.

This final step in the tone attenuation process regenerates an unweighted output signal. The signal regenerated by the IFFT in Step 16 is an analysis-window in duration and exhibits the amplitude weighting that was imposed on the input signal. As explained in Section 2.1, the upper half of one window and the lower half of the succeeding one contain the same time function data, but with reverse amplitude weightings. Therefore, by adding the upper half of the signal regenerated for the preceding analysis window to the lower half of the signal regenerated for the current analysis window, the unweighted segment of the signal, half an analysis window in duration, is recovered. It will be identical to the corresponding original input segment in all respects but two: all tones detected in the spectrum of the segment will be attenuated, as will be all signal components above 3000 Hz. The procedure in Step 18 is first to add the two half windows as described above and then to store the upper half of the current window in preparation for regenerating the next unweighted segment of the output signal.

Step 19.

The setting of the control panel switch that is used to enable the automatic impulse detection process is tested. If the stored value is negative the next step is skipped.

Step 20. (Detailed in Section 5.3.3)

As described in Section 3, amplitude thresholds are computed and are used to detect impulses. The detected impulses are deleted from the signal and the vacated space is filled to form a spectrally continuous transition across the region of the signal that formerly was occupied by the impulse.

Step 21.

The fully processed segment of the output signal is examined to find its peak signal level. If the peak level is greater than 4000 units, the output gain is reduced by 4.4 dB. If the peak level is less than 500 units the gain is increased by 1.6 dB. If the peak level is within the range 500 units to 4000 units the output gain is left as it was for the preceding segment of the output signal.

5.3 Descriptions of the Signal Processing Programs

5.3.1 The Peak Matching Program (MATCH$PK)

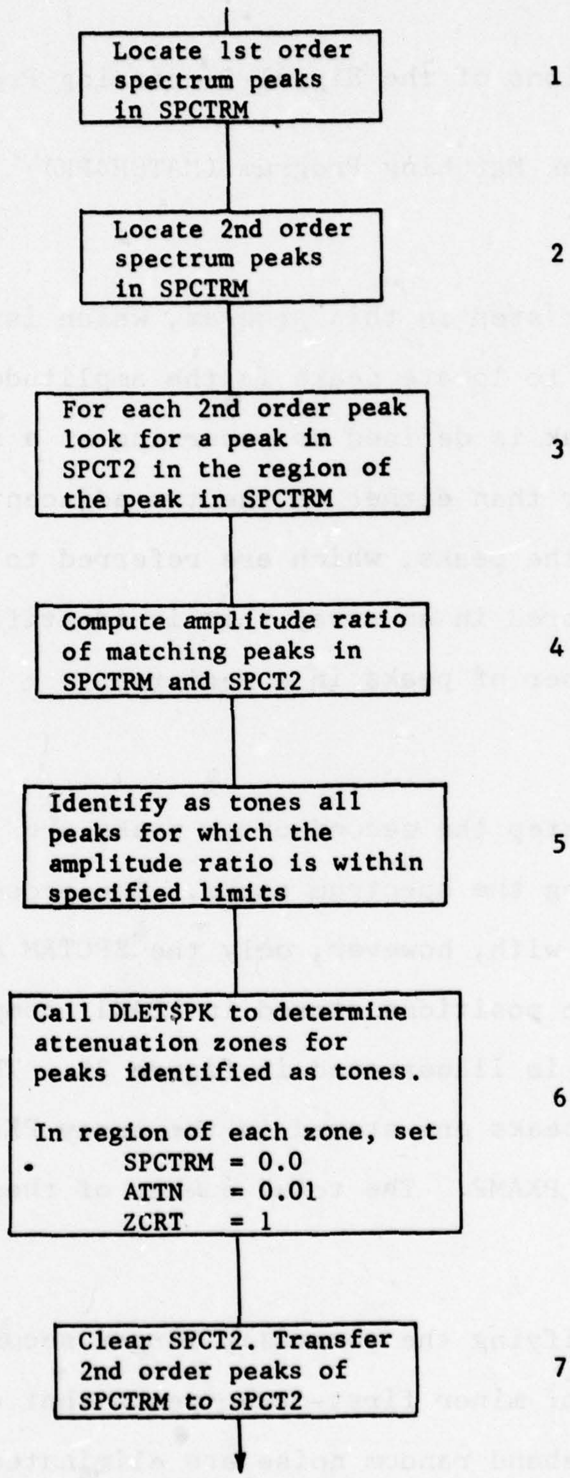
Step 1

The first step in this program, which is diagrammed in figure 25, is to locate peaks in the amplitude spectrum array SPCTRM. A peak is defined as occurring at a spectrum sample that is larger than either of the two adjacent samples. The locations of the peaks, which are referred to as first-order peaks, are stored in an array that is identified as PEAK1, and the total number of peaks in a register.

Step 2

In this step the second-order peaks are located. These are the peaks among the spectrum peaks. The procedure is the same as for step 1 with, however, only the SPCTRM samples that are located at the positions stored in PEAK1 compared in amplitude. The procedure is illustrated in figure 26. The locations of the second-order peaks are stored in the array PKLOC1, and their amplitudes in PKAMP. The total number of these peaks is stored in NPK2.

By identifying the generally larger second order peaks, the large number of minor first-order peaks that usually are associated with wideband random noise are eliminated from further consideration. The iterative peak selection process is stopped



'URE 25 SEQUENCE OF STEPS IN THE PEAK MATCHING PROGRAM

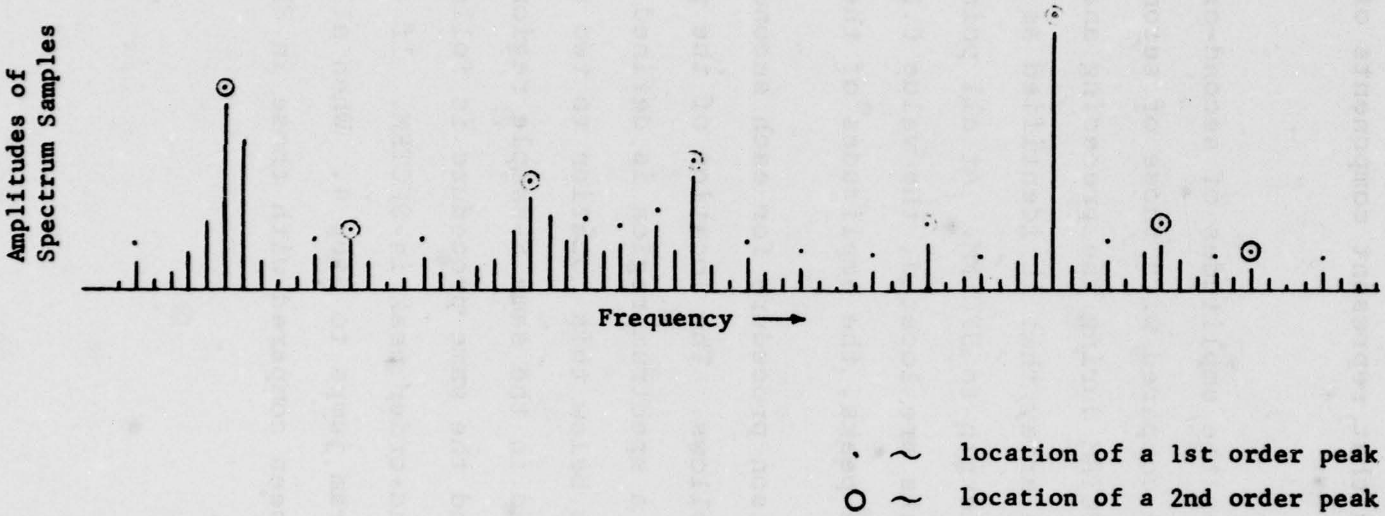


FIGURE 26 LOCATION OF SECOND-ORDER PEAKS

at this stage since carrying it further could lead to the elimination of peaks that represent components of weak tones.

Step 3

In this step the amplitudes of second-order peaks in the SPCTRM array are compared with those of second-order peaks that were found in SPCTRM during the preceding analysis window. These are stored in an array that is identified as SPCT2. SPCT2 is an array equal in length to SPCTRM. At all points except where second-order peaks are located, the value 0.0 is stored. At the locations of the peaks, the amplitudes of the peaks are stored.

The comparison procedure for each second-order peak in SPCTRM is as follows. The location of the peak is retrieved from PKLOC1 and a spectrum region is defined that ranges from two spectrum samples below this location to two samples above it. SPCT2 is searched in the same 5-sample region for non-zero values. If none are found the same procedure is followed for the next sequential second-order peak in SPCTRM. If a peak is found in SPCT2, the program jumps to Step 4. When all second-order peaks in SPCTRM have been compared with those in SPCT2, the program jumps to Step 7.

Step 4

In this step, the APU computes the ratio of the amplitudes of the peaks within the same 5-sample wide regions in SPCTRM and in SPCT2.

Step 5

The computed ratio is compared to limits that are specified as in Table 6.

TABLE 6

AMPLITUDE RATIO LIMITS FOR MATCHING SPECTRUM PEAKS

Process Period (ms)	Lower Limit	Upper Limit
50	0.93	1.07
100	0.87	1.15
200	0.76	1.32

If the ratio is not within the appropriate limits, the peak in SPCTRM is not considered to be a tone component, and the program loops back and performs Step 3 for the next sequential second-order peak in SPCTRM.

Step 6

This step is reached only when a second-order peak in SPCTRM matches a peak in SPCT2 in both location and amplitude. The program computes a threshold that is 30 dB below the amplitude of the peak in SPCTRM that is to be attenuated. It then calls on a routine that uses this threshold to determine the spectrum range that is to be attenuated in the region of the SPCTRM peak. Within the determined attenuation region, this routine (which is called DLET$PK) then sets the values stored in SPCTRM to 0, the values in FPSPCT to 0.0, the values in ATTN to 0.01, and the values in ZCRT to 1. The program then loops back and performs Step 3 for the next sequential second-order peak in SPCTRM.

Step 7

With all of the second-order peaks in SPCTRM having been examined and those found to be tones attenuated, the program prepares for the processing of the SPCTRM peaks in the next analysis window. First it clears the SPCT2 array by setting to zero all of the peaks previously stored there. Then it transfers the second-order peaks of SPCTRM to SPCT2. Since some of these were set to zero by DLET$PK, the data for the locations and amplitudes of the peaks are obtained from the arrays PKLOC1 and PKAMP.

5.3.2 The Amplitude Threshold Program (THRESHLD)

The steps that implement this procedure are diagrammed in figure 27.

Step 1

The average of all non-zero spectrum samples in FPSPCT (the floating-point amplitude spectrum) is computed. (Zero amplitude values will be found in regions of FPSPCT that were

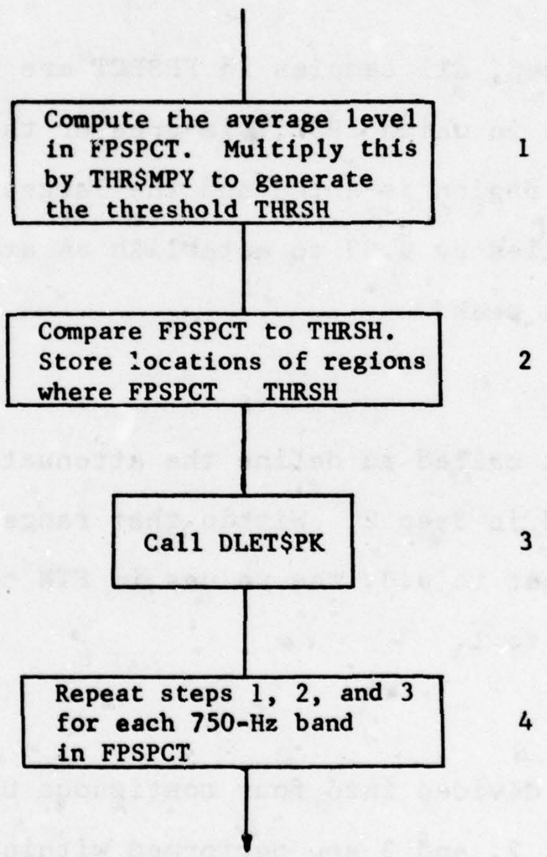


FIGURE 27 SEQUENCE OF STEPS IN THE AMPLITUDE THRESHOLD PROGRAM

attenuated during the execution of the MATCHPK routine and during the application of the manually set attenuation gates). This average is multiplied by a factor, THR$MPY, that is selected in accordance with Table 3 (2.2.2) to generate the peak detection threshold, THRSH.

Step 2

In this step, all samples in FPSPCT are compared to THRSH. For each region in which FPSPCT is greater than THRSH, the start and end of the region is noted and the largest sample in the region is multiplied by 0.03 to establish an attenuation threshold 30 dB below the peak.

Step 3

DLET$PK is called to define the attenuation range at each region detected in Step 2. Within that range, the values stored in FPSPCT are set to 0.0, the values in FTN to 0.01, and the values in ZCRT to 1.

Step 4

FPSPCT is divided into four contiguous bands, each 750 Hz wide. Steps 1, 2, and 3 are performed within each band, using the appropriate value of THR$MPY to compute the detection threshold for each band.

5.3.3 The Impulse Detection and Attenuation Program (IMP)

This routine receives as input data regenerated, unweighted segments of the time waveform of the signal. Each segment is half the selected analysis window in duration (i.e., either 25ms, 50ms, or 100ms in duration). The IMP routine processess these in 25ms long sub-segments. As shown in figure 28, the sub-segment currently being processed, CYT, is loaded into the upper half of a 512-point long buffer. A copy of the preceding sub-segment, CYT, is stored in the lower 256 points of the buffer. The steps in the IMP program are illustrated in figure 29.

Step 1

An impulse detection threshold P, is computed as five times the average of the absolute amplitudes of all 256 samples in the current sub-segment, CYT.

Step 2

The absolute amplitude of each point in CYT is compared to the threshold. All values that are greater than the threshold are set to zero.

Step 3

The region from point A (at location 128 in the buffer) through point B (at location 384) is examined for the presence of groups of zero amplitude samples. The start location and end location of each such group (which we refer to as a gap) are

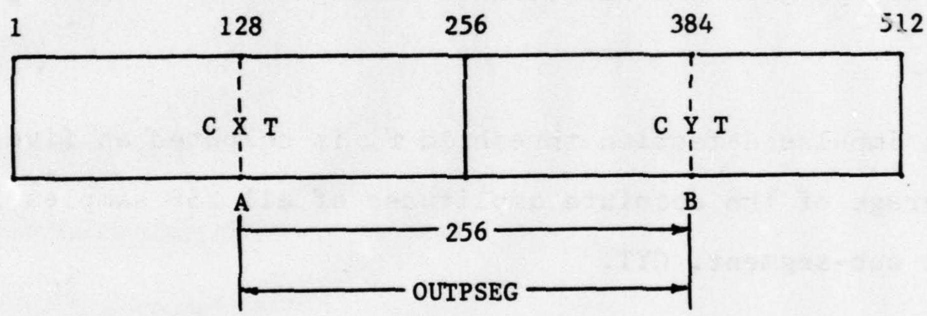


FIGURE 28 ORGANIZATION OF DATA IN THE IMPULSE PROCESSING BUFFER

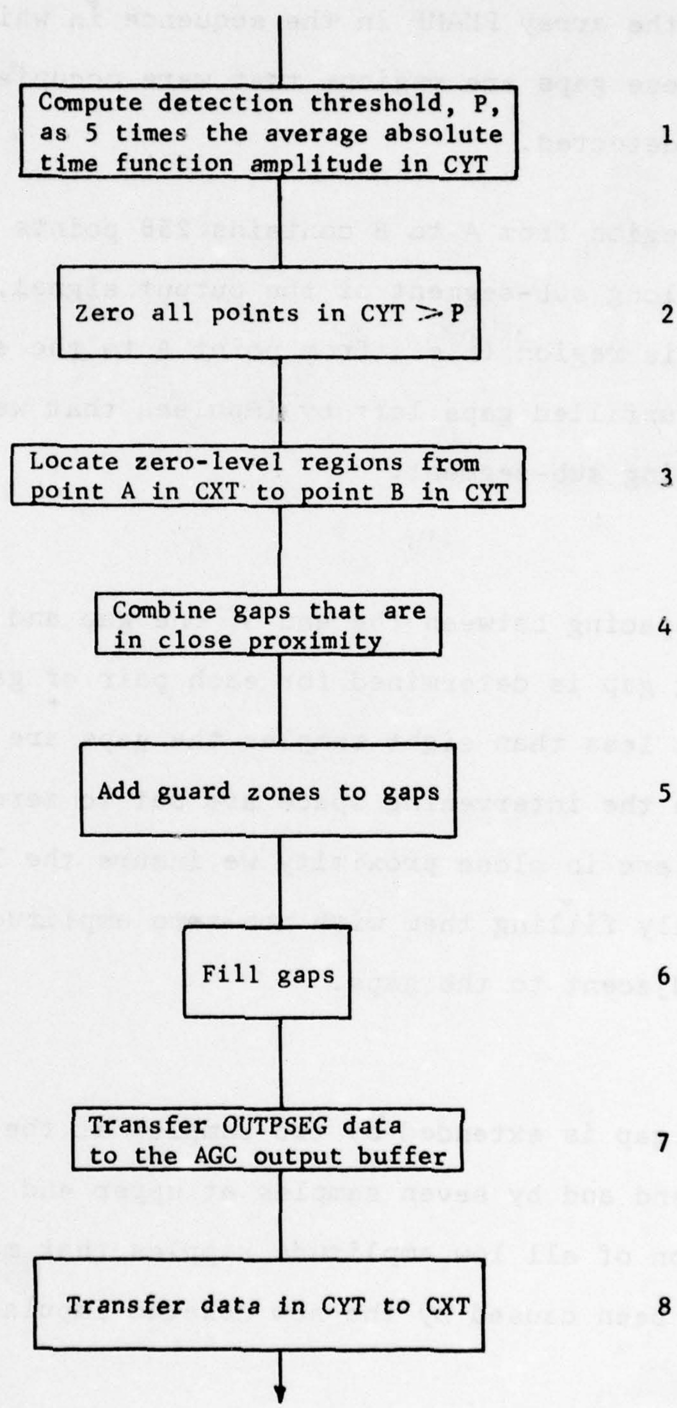


FIGURE 29 SEQUENCE OF STEPS IN THE IMPULSE PROCESSING PROGRAM

stored in the array PKAMP in the sequence in which they are found. These gaps are regions that were occupied by impulses that were detected.

The region from A to B contains 256 points and so represents a 25.6 ms long sub-segment of the output signal. The lower half of this region (i.e., from point A to the end of CYT) contains the unfilled gaps left by impulses that were deleted in the preceding sub-segment.

Step 4

The spacing between the end of one gap and the start of the succeeding gap is determined for each pair of gaps. Where this spacing is less than eight samples the gaps are combined and the samples in the intervening space are set to zero. By combining gaps that are in close proximity we insure the likelihood of successfully filling them with non-zero amplitude samples in the regions adjacent to the gaps.

Step 5

Each gap is extended by two samples at the lower (i.e., earlier) end and by seven samples at upper end to insure the elimination of all low amplitude ripples that may have accompanied or been caused by the now deleted impulses.

Step 6

Each gap in the region from point A to point B is filled in the order in which the gaps occur. The procedure as described in Section 3, is as follows. Regions equal in width to the gap are defined for the non-zero segments of signal that immediately preceed and immediately follow the gap. The time-waveform samples in these regions are weighted, rotated, and combined as is illustrated in figure 14. A gap that begins at point A can use the entire region from that point to the start of the buffer for filler material. Likewise, a gap ending at point B can use the region that extends to the end of the buffer. Since these regions each contain 128 samples, the duration of the largest gap that can be completely filled by this procedure is 12.8 ms.

Step 7

The time function data in the region from point A through point B are transmitted to the next routine, for automatic adjustment of the level of the output signal.

Step 8

The data stored in CYT are shifted into CXT in preparation for processing the next 25.6 ms sub-segment of the regenerated signal.

6.0 SUMMARY AND RECOMMENDATIONS

The Speech Enhancement Advanced Development Model that is described in this report satisfies the requirements that were imposed on it both by the contract under thich it was developed and by the practical conditions under which it was to be used. It meets all design and performance objectives, exceeding many of them by a comfortable margin. At the same time, it is easy to use and reasonably easy to move from location to location. Once it had been shown that the ADM met all the objective tests of its performance, it remained only to test its value as an adjunct to a speech communication system. Its performance in this regard was evaluated by trained Air Force personnel who used the system under realistic conditions during the normal performance of their duties. Speech signals that had been received over noisy communication channels were processed by the ADM and the effect of the ADM on their intelligibility and quality evaluated, as was the convenience and ease of use of the system. The results of these tests showed that, with few exceptions, the system was effective in improving the intelligibility and quality of received speech signals.

After analyzing the evaluating the tests results, the RADC technical monitors of this contract recommended that several changes be made in the ADM. All of them are based on the observation that the users of the ADM made minimum use of the manual controls in order to concentrate on performing their main tasks.

The first recommended change was that the manual control of the input signal level be replaced by an automatic control, such as an AGC circuit. This change was being contemplated even before the system tests had been started. If a multiplying DAC is used to adjust the signal level, the input level could be varied automatically so as to make maximum use of the dynamic range of the input unit. To illustrate, when the impulse detection process (IMP) is enabled, the maximum dynamic range will be achieved when the average signal level is slightly less than one-fifth of the permissible peak input to the A/D converter. Since the peak input results in an A/S output of 8192 units, the optimum average level of the input signal is at an A/D level of about 1630 units. At this average level, the IMP program can still detect the presence of impulse. Alternatively, when the IMP process is not enabled, the signal level should be adjusted such that the peak signal level is at the peak input level (i.e., 8192 units). Since the system cannot know in advance what either the average or maximum levels will be in any analysis window, control of the DAC must be based on measurements of the signal level made during the analysis window that precedes the one for which the signal level is to be adjusted. To provide a margin for change in signal level from window-to-window, the DAC gain could be set at some fraction of the value that otherwise would be ideal. At the output of the ADM, the signal level could be adjusted, before applying the output AGC, to restore the original dynamic variations of the input signal, that is, to negate the effects of the input AGC. Equally important, these effects would have to be taken into account by the routine that matches the amplitudes of peaks on adjacent amplitude spectrums.

The second recommended change was the removal of the manual process for detecting tones and for setting attenuation gates. In large part, this recommendation reflects the effectiveness of the automatic tone detection process. Removal of the manual process would permit considerable simplification of the ADM. More than half of the controls on the control panel could be eliminated, leaving only the output signal level, process period, and automatic process-enable controls. In addition, the display oscilloscope would no longer be needed, nor would the circuitry and software that generate the display waveforms. The resulting simplifications would reduce the height, weight, and power requirements of the ADM.

In a different area of proposed change, it was observed that users of the ADM tended to leave the process period set to 200 ms at all times. In view of this, it was suggested that the process period be fixed at this value. However, it may be desirable to retain some flexibility in the setting of this parameter. Accordingly, it is recommended that the choice of process period be limited to two selections; long (i.e., 200 ms) and short (50 ms).

Finally, a major change can be made in the system for loading the ADM programs into the MAP. The Linc Tape Unit and the PDP 11/04 minicomputer were appropriate components of this system while the ADM software was being developed. Since the time when these units were acquired, significant advances have been made in semiconductor firmware memories. In particular, EPROM's are now available that can store 2048 bytes on a single chip. The current version of the ADM software could be stored on four EPROM chips on a memory board inside the MAP. Even with the addition of the INTEL program, and of other as yet unspecified signal processing programs, it is likely that an eight-chip memory would satisfy all future needs of the ADM.

Several major improvements would result from the elimination of the tape reader and the minicomputer. First, and most obvious, is a substantial reduction in the height, weight, and power requirements of the ADM. Second, the overall reliability of the ADM would be improved since both the tape unit and the minicomputer have been, and are likely to continue to be, sources of failures in the system. Finally, the loading of the program from the EPROM memory could be arranged to occur automatically upon the application of power to the MAP. This would eliminate the need for the user to press a LOAD button to initiate operation of the ADM. Thus, even in the event of a power failure, the ADM would resume operation immediately upon the restoration of power.

Taken together, the recommended changes would result in an almost fully automatic speech enhancement ADM that was smaller, lighter, and more reliable. We estimate that if all the recommended changes were made, the ADM would be less than 20 inches high, weigh less than 110 pounds, and draw about 10 amperes at 110 volts.

APPENDIX

GENERAL DESCRIPTION OF THE MAP SYSTEM

A.1 Map Configuration

The Macro Arithmetic Processor (MAP) System consists of several types of programmable processors and the associated memory and peripheral units required for the particular application under consideration. A Central processor Unit (CSPU) functions as an executive controller. In its role of resource management, it interprets host commands, transfers programs from MAP memory to the various MAP processors, sequences their program execution, and performs incidental processing. The Arithmetic Processor (AP) accesses data in the MAP memory, performs floating point arithmetic calculations, and returns results to MAP memory. The Host Interface Scroll (HIS) effects transfers of data and commands between the host and MAP memory and permits the host to load data and programs into MAP memory. An Input/Output Scroll (IOS) controls block transfer to or from external peripheral devices and can transfer data into or out of the MAP memory without interfering with the processing cycle.

Figure 30 illustrates the interconnection relationship between the various processors and memories of the MAP system.

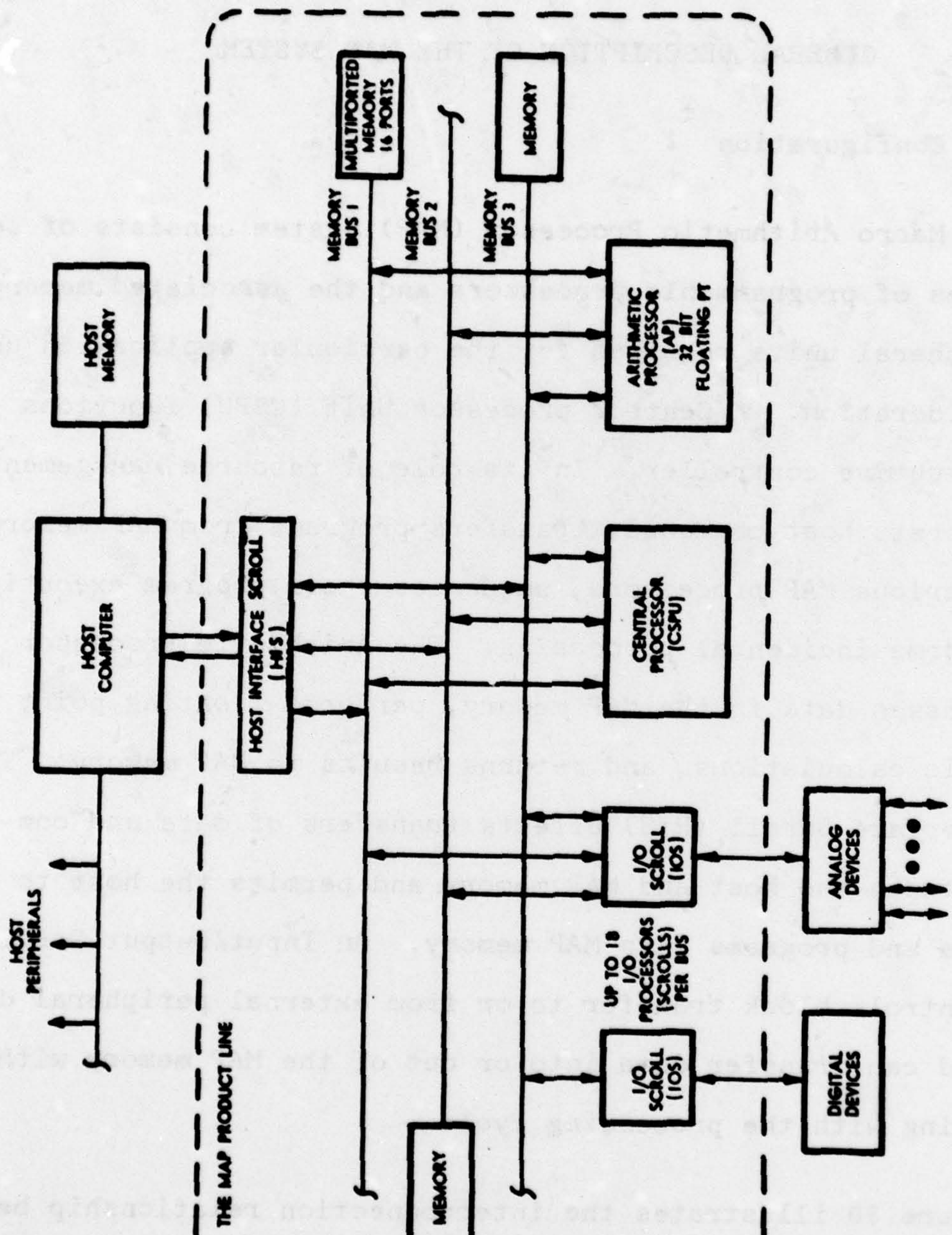


FIGURE 30 MAP CONFIGURATION

The CSPU, AP, HIS, and IOS all operate asynchronously and simultaneously. Except for the CSPU, they use program routines stored in small memories associated with each processor. The CSPU uses routines stored in MAP memory. In its role of resource management the CSPU sees that blocks of program instructions are supplied to the other processor memories as needed.

Memory serves as a common ground of communication for processors since all have access to some MAP memory. Moreover, various registers within peripheral interfaces and processors are assigned addressed within the main MAP memory. Such "pseudo" memory registers (PMR's) may be read-only or write-only.

All data and instruction transfers within MAP, between memory and the various processors, are over three common memory busses. The memory controller for each bus assigns cycles on a priority basis to each device or processor requesting use of the bus.

The programs for the CSPU, AP, HIS, and IOS are defined separately. Moreover, the addressing/indexing part of the AP function is handled by a subprocessor of the AP called the AP Addresser (APS), while the arithmetic part is handled by a subprocessor called the AP Arithmetic Unit (APU). These two subprocessors are separately programmed greatly simplifying the programming task and making future modifications easy.

A.2 Central Processing Unit (CSPU)

The CSPU module plays the role of resource management in the multiprocessor system. It is the single controlling processor in the MAP. It initiates processing sequences and controls data flow paths in the system. It has a fast fixed-point arithmetic unit for address calculations for both data and program instruction modules, an eight register accumulator file, an instruction register, and a priority interrupt network. It is, in effect, a minicomputer in its own right with a large repertoire of instructions.

The CSPU has access to the main memories in MAP. After the host computer transfers the MAP processing routines (for all MAP processors) to MAP memory, the CSPU proceeds through its executive program to control the operation of the other MAP processors by supplying them with blocks of program instructions from MAP memory. Each processor, other than the CSPU, has its own associated memory for storage of its program instructions.

The CSPU control of the Arithmetic Processor permits instruction command sequences and data memory addresses to be enterer into the AP. If the AP indicates by means of system flags that it is busy, the CSPU will idle until the AP is ready for new instructions. Alternatively, the AP can generate an interrupt to the CSPU if error conditions exist or upon completion of an operation.

The CSPU control of the I/O Scrolls extends to initializing the IOS with a base memory address, difference pattern parameters, and a block size parameter. Then, to carry out a complete input/output operation via an IOS, the CSPU merely transfers a command word to the pseudo-memory register of the appropriate IOS.

A.3 Arithmetic Processor (AP)

The AP performs all of the computations directly associated with a processing algorithm. Its internal program routines and general guidance are initially supplied by the CSPU; but the execution of its programs is relatively independent of all other processors in the MAP.

Figure 31 illustrates the organization of the AP. The AP is divided, both functionally and physically, into sub-processors. The Arithmetic Processor Unit (APU) carries out the computation of the programmed algorithm without regard to the details of data addressing and indexing. The Arithmetic Control Module (ACM) contains the memory for the APU and contains the entire Arithmetic processor Addresser (APS). The Arithmetic Processor Addresser (APS) computes the address in MAP memory of both input data words to be processed by the APU and output locations for subsequent results. Both the APU and APS have their own associated memories for storage of their parallel but independent

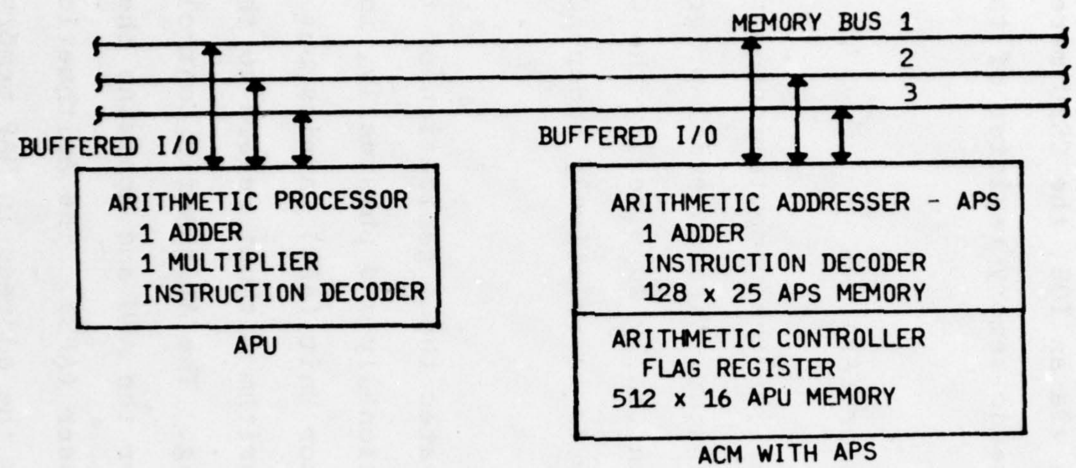


FIGURE 31 SIMPLIFIED ORGANIZATION OF THE ARITHMETIC PROCESSOR (AP) IN THE MAP 200

programs. Communications between these two subprocessors is by means of system flags, implemented in hardware.

The APU executes a stored program routine using its own memory to carry out the arithmetic algorithm. Input data is buffered from MAP memory in an Input Data Queue (IQ). Mathematical operations are distributed among the arithmetic adders and multipliers with intermediate results stored in several available registers. Upon completion of the processing, results are buffered into MAP memory via Output Data Queues (OQ). These queues are FIFO (First-In-First-Out) systems.

The MAP-100, -200, or -300 models, in order of increasing capability, each have a different configuration of the AP.

The APU in the MAP-300 contains two adders, two multipliers, 34 registers, and three high-speed FIFO buffers (one input data queues and two output data queues). The arithmetic units themselves work in parallel with the mathematic operations allocated between the two multipliers and two adders. The speed of operation of the MAP-300 is likely to be balanced between memory limited and processor limited when bipolar memory is used for data.

The APU in the MAP-200 has half the multipliers, adders, registers and input/output queues. It is therefore arithmetic limited when the bipolar IC memory is used; however, it is usually memory limited when the slower MOS memory is used.

The arithmetic addresser executes its own stored program to calculate addresses of data words for use by the APU. The APS contains its own memory, arithmetic unit and program controller. Addresses are calculated and buffered to the MAP memory busses via two FIFO (First-In-First-Out) systems. Addresses of input data words for the APU are placed in the APS Read Address FIFO (RAF). Addresses for APU results are placed in the APS Write Address FIFO (WAF).

A.4 MAP Memory

The MAP System Memory is available in either an ultra-fast bipolar integrated circuit version with a 125 nanosecond access time or in a slower MOS version with a 500 nanosecond access time. Both types are configured to use 32-bit words to be compatible with the floating point format of the MAP and they are addressable in bytes, half-word, and full-word increments to optimize I/O efficiency and storage economy for different types of problems and I/O devices.

Memory architecture is such that there are three independent memory busses each capable of accommodating either 256K bytes of addressable MOS memory, or 64K bytes of addressable bipolar memory. Since multiple memories may be employed and since each bus operates independently of the others, it is possible for several processors to each access a memory simultaneously and continuously without conflict with another processor. For ultra-high-speed processing situations it is possible to almost completely overlap input, output, and processing operations with all processors performing at peak efficiency thereby precluding the inefficiencies inherent in serial systems.

The memory is multiported with up to 11 processor ports per bus. The HIS accounts for a single port, the CSPU accounts for one port, and the AP accounts for one port in MAP-100 or MAP-200 and two ports in MAP-300. Therefore, there are at least 7 ports available per bus for I/O Scrolls.

For each of the three busses there is one master memory control module. If two devices simultaneously attempt to access a single memory, arbitration of priorities is accomplished within the appropriate master memory control module. Two general priority levels have been established: one is an absolute high priority level required by synchronous devices, such as magnetic tapes or discs with minimal buffering; the other is a sequential network which takes slower devices or buffered MAP modules in a pre-established order.

The memory bus structure is one of the features distinguishing the MAP from other array processors. The MAP is a memory-centered device. All instruction and data transfers within MAP are via the three parallel, independent memory busses. The three master memory modules each control information flow on their own bus. Thus, the memory is the center of MAP information transfer and control.

A.5 I/O Scrolls (IOS)

The I/O Scrolls (IOS) are input/output processors which transfer data between MAP memory and external devices in a preprogrammed manner established by the CSPU. There are three basic versions of the IOS which vary in technical complexity and throughput speed. All of the IOS models are predicated on a difference pattern generator in which the primary parameter is the extent of the address space in memory between adjacent data words of interest.

The IOS/1 is a single-word I/O handler. It is most often used with slow devices, such as teletypes, tape and card readers, etc. There is one byte, half-word, of full-word transferred for each instruction from the CSPU to the IOS. Maximum I/O transfer rate for the IOS/1 is about 5 kilobytes per second.

In computer systems that do not have a MAP, input/output operations which require demultiplexing of data streams or possibly even more complex addressing patterns are frequently done on a programmed basis by the computer itself or by a special purpose preprocessor which may even be another computer. Programmed I/O operations create a significant overhead penalty by having the computer calculate the appropriate data address. This time burden not only impairs the overall processing throughput rate but may also eliminate the possibility of high speed data transfers.

The IOS/2 and /2F overcome these drawbacks for I/O operations which involve cyclic addressing schemes. A one-word command to the IOS initiates an I/O sequence where sequential data words are input/output to memory locations with address differences which follow a specified pattern. This sequence runs without further attention from the CSPU. These patterns are stored in a small IOS memory by the CSPU. Thus, the IOS need receive only a single command from the CSPU to start, and continue until completion, a block demultiplexing I/O routine with a predefined pattern.

MISSION
of
Rome Air Development Center

RADC plans and conducts research, exploratory and advanced development programs in command, control, and communications (C^3) activities, and in the C^3 areas of information sciences and intelligence. The principal technical mission areas are communications, electromagnetic guidance and control, surveillance of ground and aerospace objects, intelligence data collection and handling, information system technology, ionospheric propagation, solid state sciences, microwave physics and electronic reliability, maintainability and compatibility.

AMERICAN REVOLUTION BICENTENNIAL 1776-1976